Clustering via Diffusion Adaptation over Networks

Xiaochuan Zhao and Ali H. Sayed

Department of Electrical Engineering
University of California, Los Angeles

***Abstract*—Distributed processing over networks relies on in-network processing and cooperation among neighboring agents. Cooperation is beneficial when all agents share the same objective or belong to the same group. However, if agents belong to different clusters or are interested in different objectives, then cooperation can be damaging. In this work, we devise an adaptive combination rule that allows agents to learn which neighbors belong to the same cluster and which other neighbors should be ignored. In doing so, the resulting algorithm enables the agents to identify their grouping and to attain improved learning and estimation performance over networks.**

***Index Terms*—Diffusion adaptation, clustering, diffusion LMS, combination weights, energy conservation.**

I. Introduction

Several strategies that enable distributed optimization and estimation over networks were proposed and studied in the literature, such as consensus strategies [1], [2], incremental strategies [3]–[6], and diffusion strategies [7], [8]. Among them, diffusion strategies are scalable, robust, and able to endow networks with real-time adaptation and learning abilities. They were successfully applied to model various forms of complex and self-organized behavior in biological networks [9], [10] and to solve general optimization problems [11].

In most of these earlier studies, agents were seeking a common objective, such as finding the global minimizer for a cost function. There are important situations where different agents in a network are interested in different objectives [12]–[19]. These situations arise frequently in clustering problems where a subset of the agents belongs to one group and another subset belongs to a second group. Solving distributed estimation problems that involve multiple clusters is challenging because agents first need to figure out which subset of their neighbors have the same objective as their own; otherwise, cooperation among agents with different objectives may lead to catastrophic results (see the simulation results in Fig. 4b).

In this work, we show how to design the combination weights adaptively such that agents are able to cluster and cooperate only with neighbors that share the same objective. We formulate an optimization problem that suggests a particular construction for the weights. Then, we examine the resulting behavior by simulation and theory.

A. Notation

We use lowercase letters to denote vectors, uppercase letters for matrices, plain letters for deterministic variables, and boldface letters for random variables. We also use $(\cdot)^*$ to denote conjugate transposition, $(\cdot)^{-1}$ for matrix inversion, $\mathrm{Tr}(\cdot)$ for the trace of a matrix, $\otimes$ for Kronecker products, and $\rho(\cdot)$ for the spectral radius of a matrix. All vectors in our treatment are column vectors, with the exception of the regression vectors, $\boldsymbol{u}_{k,i}$, which are taken to be row vectors for convenience of presentation.

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

II. Networks with Multiple Clusters

We consider a connected network consisting of N nodes. Each node k collects scalar measurements $\boldsymbol{d}_k(i)$ and $1 \times M$ regression data vectors $\boldsymbol{u}_{k,i}$ over successive time instants $i \geq 0$. The measurements across all nodes are assumed to be related to a set of unknown $M \times 1$ vectors $\{w_k^o\}$ via a linear regression model of the form [20]:

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

where $\boldsymbol{v}_k(i)$ denotes measurement or model noise and w_k^o denotes the parameter of interest for node k. For example, w_k^o can be the parameter vector of some underlying physical phenomenon, the location of a food source, or a vector modeling different groupings of nodes. The nodes in the network would like to estimate the vectors $\{w_k^o\}$ by seeking the solution for the following minimization problem:

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

In our previous works [7], [8], a common value for all vectors, i.e., $\{w_k^o = \theta^o\}$, was assumed so that all nodes across the network were pursuing the same unknown parameter vector θ^o. Through in-network processing and local cooperation with their neighbors, nodes were able to estimate θ^o adaptively by means of diffusion strategies.

However, when there exist multiple unknown model vectors $\{\theta_m^o; m > 1\}$, it becomes more challenging to enforce meaningful cooperation among nodes. This is because nodes do not know beforehand whether they are sensing information originating from one model or another. They also do not know which models are influencing the data received from their neighbors. If nodes process these data regardless of the models by which they were generated, then the resulting estimates will likely be distorted.

Without loss of generality, let us assume that there are only two possible models, say, θ_1^o and θ_2^o. Nodes affected by θ_1^o are collected into the set $\mathcal{N}^{(1)}$ and nodes influenced by θ_2^o are

978-1-4673-1878-5/12/$31.00 ©2012 IEEE

collected in the set $\mathcal{N}^{(2)}$ — in this way, the original network is partitioned into two non-overlapping subsets $\mathcal{N}^{(1)}$ and $\mathcal{N}^{(2)}$. Then, the cost function (2) can be decoupled into two separate problems:

$$\underset{\theta_m}{\text{minimize}} \sum_{k\in\mathcal{N}^{(m)}} \mathbb{E}|\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\theta_m|^2 \tag{3}$$

for $m = 1, 2$. In principle, each of these problems can be solved separately by using the Adapt-then-Combine (ATC) diffusion LMS strategy of [8] as follows:

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} + \mu_k \boldsymbol{u}_{k,i}^* \left[\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{w}_{k,i-1}\right] \tag{4}$$

$$\boldsymbol{w}_{k,i} = \sum_{l\in\mathcal{N}_k\cap\mathcal{N}^{(m)}} a_{lk}(i)\boldsymbol{\psi}_{l,i} \tag{5}$$

where $k \in \mathcal{N}^{(m)}$ and $\mathcal{N}_k$ denotes the neighborhood of node k in the original network. In (5), the coefficients $\{a_{lk}(i)\}$ are nonnegative entries of an $N \times N$ combination matrix A_i at time i. The coefficients $\{a_{lk}(i)\}$ are zero whenever node l is not connected to node k or node l is pursuing a different objective from node k is, i.e., $l \notin \mathcal{N}_k \cap \mathcal{N}^{(m)}$. We require the matrix A_i to be left-stochastic, i.e., $A_i^{\mathsf{T}}\mathbb{1}_N = \mathbb{1}_N$, where $\mathbb{1}_N$ denotes the $N \times 1$ vector with all entries equal to one.

Comparing (4)–(5) with the traditional ATC diffusion strategy from [8], namely,

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} + \mu_k \boldsymbol{u}_{k,i}^* \left[\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{w}_{k,i-1}\right] \tag{6}$$

$$\boldsymbol{w}_{k,i} = \sum_{l\in\mathcal{N}_k} a_{lk}(i)\boldsymbol{\psi}_{l,i} \tag{7}$$

we see that the key difference lies in the composition of the neighborhoods of node k in steps (5) and (7): neighbors of node k that are seeking another model are excluded from the combination step (5). The challenge in implementing (5) therefore lies in developing a procedure that would enable node k to learn which of its neighbors should be excluded from the combination (7) to obtain (5).

We explain in the sequel that a diffusion strategy of the form (4)–(5) can be achieved by using a traditional strategy of the form (6)–(7) and by designing the coefficients $\{a_{lk}(i)\}$ in (7) in such a manner that they assume relatively small values for neighbors that are pursuing a different objective. Specifically, we develop an adaptive procedure for adjusting these combination weights so that nodes in the network can learn on the fly which nodes should be excluded from their neighborhoods.

III. Mean-Square Performance Analysis

Before we explain how to optimize the combination matrix A_i to enable clustering, we first examine the performance of the algorithm as a function of these combination weights.

A. Error Recursion

We introduce the error vectors at each node k:

$$\widetilde{\boldsymbol{\psi}}_{k,i} \triangleq w_k^o - \boldsymbol{\psi}_{k,i}, \qquad \widetilde{\boldsymbol{w}}_{k,i} \triangleq w_k^o - \boldsymbol{w}_{k,i} \tag{8}$$

where w_k^o is either θ_1^o or θ_2^o. We substitute the linear regression model (1) into (6)–(7) to get

$$\widetilde{\boldsymbol{\psi}}_{k,i} = (I_M - \mu_k \boldsymbol{R}_{k,i})\,\widetilde{\boldsymbol{w}}_{k,i-1} - \mu_k \boldsymbol{s}_{k,i} \tag{9}$$

$$\widetilde{\boldsymbol{w}}_{k,i} = \sum_{l\in\mathcal{N}_k} a_{lk}(i)\widetilde{\boldsymbol{\psi}}_{l,i} + w_k^o - \sum_{l\in\mathcal{N}_k} a_{lk}(i) w_l^o \tag{10}$$

where I_M denotes the $M \times M$ identity matrix and

$$\boldsymbol{R}_{k,i} \triangleq \boldsymbol{u}_{k,i}^*\boldsymbol{u}_{k,i}, \qquad \boldsymbol{s}_{k,i} \triangleq \boldsymbol{u}_{k,i}^*\boldsymbol{v}_k(i) \tag{11}$$

We collect the various quantities from across all nodes into the following block vectors and matrices:

$$\boldsymbol{\mathcal{R}}_i \triangleq \operatorname{diag}\{\boldsymbol{R}_{1,i}, \boldsymbol{R}_{2,i}\ldots, \boldsymbol{R}_{N,i}\} \tag{12}$$

$$\boldsymbol{s}_i \triangleq \operatorname{col}\{\boldsymbol{s}_{1,i}, \boldsymbol{s}_{2,i}\ldots, \boldsymbol{s}_{N,i}\} \tag{13}$$

$$\mathcal{M} \triangleq \operatorname{diag}\{\mu_1 I_M, \mu_2 I_M\ldots, \mu_N I_M\} \tag{14}$$

$$w^o \triangleq \operatorname{col}\{w_1^o, w_2^o\ldots, w_N^o\} \tag{15}$$

$$\widetilde{\boldsymbol{\psi}}_i \triangleq \operatorname{col}\left\{\widetilde{\boldsymbol{\psi}}_{1,i}, \widetilde{\boldsymbol{\psi}}_{2,i}\ldots, \widetilde{\boldsymbol{\psi}}_{N,i}\right\} \tag{16}$$

$$\widetilde{\boldsymbol{w}}_i \triangleq \operatorname{col}\{\widetilde{\boldsymbol{w}}_{1,i}, \widetilde{\boldsymbol{w}}_{2,i}\ldots, \widetilde{\boldsymbol{w}}_{N,i}\} \tag{17}$$

From (9)–(10), the recursion for the block error vector $\widetilde{\boldsymbol{w}}_i$ is found to be:

$$\boxed{\widetilde{\boldsymbol{w}}_i = \mathcal{A}_i^{\mathsf{T}}(I_{NM} - \mathcal{M}\boldsymbol{\mathcal{R}}_i)\widetilde{\boldsymbol{w}}_{i-1} - \mathcal{A}_i^{\mathsf{T}}\mathcal{M}\boldsymbol{s}_i + b_i} \tag{18}$$

where

$$\mathcal{A}_i \triangleq A_i \otimes I_M \tag{19}$$

$$b_i \triangleq \operatorname{col}\{b_{1,i}, \ldots, b_{N,i}\} = \left(I_{NM} - \mathcal{A}_i^{\mathsf{T}}\right) w^o \tag{20}$$

The entries of the block vector b_i can be interpreted as penalty terms for choosing the wrong neighbors since

$$b_{k,i} = w_k^o - \sum_{l\in\mathcal{N}_k} a_{lk}(i) w_l^o \neq 0 \tag{21}$$

if there exists some neighbor $n \in \mathcal{N}_k\backslash\{k\}$ such that $a_{nk}(i) > 0$ and $w_n^o \neq w_k^o$.

B. Variance Relation

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

Assumption 1 (Statistical properties):

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

Using energy conservation arguments [20], we can establish that

$$\boxed{\mathbb{E}\|\widetilde{\boldsymbol{w}}_i\|_{\Sigma_i}^2 = \mathbb{E}\|\widetilde{\boldsymbol{w}}_{i-1}\|_{\Sigma_{i-1}}^2 + \operatorname{Tr}(\mathcal{Z}_i\Sigma_i)} \tag{22}$$

where Σ_i is an $NM \times NM$ positive semi-definite Hermitian matrix that we are free to choose at time i, and

$$\Sigma_{i-1} \triangleq \mathcal{B}_i^* \Sigma_i \mathcal{B}_i + O(\mathcal{M}^2) \quad (23)$$

$$\mathcal{Z}_i \triangleq \mathcal{Y}_i + \mathcal{B}_i(\mathbb{E}\widetilde{\boldsymbol{w}}_{i-1})b_i^* + b_i(\mathbb{E}\widetilde{\boldsymbol{w}}_{i-1})^*\mathcal{B}_i^* + b_i b_i^* \quad (24)$$

$$\mathcal{Y}_i \triangleq \mathcal{A}_i^{\mathsf{T}} \mathcal{M}\mathcal{S}\mathcal{M}\mathcal{A}_i \quad (25)$$

$$\mathcal{B}_i \triangleq \mathcal{A}_i^{\mathsf{T}} \left(I_{NM} - \mathcal{M}\mathcal{R}_u\right) \quad (26)$$

$$\mathcal{R}_u \triangleq \mathbb{E}\boldsymbol{\mathcal{R}}_i = \mathrm{diag}\left\{R_{u,1}, \ldots, R_{u,N}\right\} \quad (27)$$

$$\mathcal{S} \triangleq \mathbb{E}\boldsymbol{s}_i\boldsymbol{s}_i^* = \mathrm{diag}\left\{\sigma_{v,1}^2 R_{u,1}, \ldots, \sigma_{v,N}^2 R_{u,N}\right\} \quad (28)$$

where the notation $O(\mathcal{M}^2)$ denotes a term whose value is on the order of $\mathcal{M}^2$ and, hence, can be ignored for sufficiently small step-sizes. Note that the recursion for Σ_{i-1} runs backward in time. Once Σ_i is chosen at time i, all previous $\{\Sigma_j\}$ for $-1 \le j \le i-1$ are determined via (23). Iterating backward over i, the weighted variance relation (22) gives

$$\mathbb{E}\|\widetilde{\boldsymbol{w}}_i\|_{\Sigma_i}^2 = \mathbb{E}\|\widetilde{\boldsymbol{w}}_{-1}\|_{\Sigma_{-1}}^2 + \sum_{j=0}^{i} \mathrm{Tr}(\mathcal{Z}_j \Sigma_j) \quad (29)$$

where, from (23),

$$\Sigma_j \triangleq \mathcal{B}_{j+1}^* \mathcal{B}_{j+2}^* \ldots \mathcal{B}_i^* \Sigma_i \mathcal{B}_i \ldots \mathcal{B}_{j+2}\mathcal{B}_{j+1} + O(\mathcal{M}^2) \quad (30)$$

for $-1 \le j \le i-1$.

C. Mean-Square Stability

Let x be a vector consisting of N blocks of size $M \times 1$ each, i.e., $x = \mathrm{col}\{x_1, x_2, \ldots, x_N\}$, $x_k \in \mathbb{C}^{M\times 1}$, and let $\|\cdot\|_2$ denote the 2-norm of its vector argument. Then, the block maximum norm of x, denoted by $\|x\|_{b,\infty}$, is defined as [21]:

$$\|x\|_{b,\infty} \triangleq \max_{1\le k\le N} \|x_k\|_2 \quad (31)$$

The induced block maximum norm of a matrix X, denoted by $\|X\|_{b,\infty}$, is defined as:

$$\|X\|_{b,\infty} \triangleq \max_{x\neq 0} \frac{\|Xx\|_{b,\infty}}{\|x\|_{b,\infty}} \quad (32)$$

It was shown in [21] that for left-stochastic matrices A_j,

$$\|\mathcal{B}_j\|_{b,\infty} \le \rho(I_{NM} - \mathcal{M}\mathcal{R}_u) \quad (33)$$

Therefore, we can choose sufficiently small step-sizes to ensure that the norms $\|\mathcal{B}_j\|_{b,\infty}$ are uniformly smaller than one for all j.

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

Let $\|X\|_*$ denote the nuclear norm of a matrix $X \in \mathbb{C}^{m\times n}$, which is defined as [22]:

$$\|X\|_* \triangleq \sum_{k=1}^{\min\{m,n\}} \sigma_k \quad (34)$$

where $\{\sigma_k\}$ are the singular values of X. It can be verified that $\|X\|_* = \|X^*\|_*$. Moreover, for any Hermitian and positive semi-definite matrix X, we have $\|X\|_* = \mathrm{Tr}(X)$. Then, from (30), (33), and Assumption 2, we get for $j \le i$:

$$\begin{aligned}
\|\Sigma_j\|_* &\le \|\mathcal{B}_{j+1}^*\mathcal{B}_{j+2}^* \ldots \mathcal{B}_i^*\|_* \cdot \|\Sigma_i\|_* \cdot \|\mathcal{B}_i \ldots \mathcal{B}_{j+2}\mathcal{B}_{j+1}\|_* \\
&= \|\mathcal{B}_i \ldots \mathcal{B}_{j+2}\mathcal{B}_{j+1}\|_*^2 \cdot \|\Sigma_i\|_* \\
&\le c^2 \cdot \|\mathcal{B}_i \ldots \mathcal{B}_{j+2}\mathcal{B}_{j+1}\|_{b,\infty}^2 \cdot \|\Sigma_i\|_* \\
&\le c^2 \cdot \|\mathcal{B}_i\|_{b,\infty}^2 \cdots \|\mathcal{B}_{j+2}\|_{b,\infty}^2 \cdot \|\mathcal{B}_{j+1}\|_{b,\infty}^2 \cdot \|\Sigma_i\|_* \\
&\le c^2 \cdot [\rho(I_{NM} - \mathcal{M}\mathcal{R}_u)]^{2(i-j)} \cdot \|\Sigma_i\|_* \quad (35)
\end{aligned}$$

where c is some positive scalar such that $\|X\|_* \le c\|X\|_{b,\infty}$ because $\|X\|_*$ and $\|X\|_{b,\infty}$ are submultiplicative norms and all such norms are equivalent [23]. It follows from (35) that, for $\|\Sigma_\infty\|_* < \infty$, the weighting matrix Σ_{-1} tends to zero because

$$\|\Sigma_{-1}\|_* \le \lim_{i\to\infty} c^2[\rho(I_{NM} - \mathcal{M}\mathcal{R}_u)]^{2(i+1)}\|\Sigma_i\|_* = 0 \quad (36)$$

Therefore, in steady-state, the weighted variance relation (29) becomes

$$\lim_{i\to\infty} \mathbb{E}\|\widetilde{\boldsymbol{w}}_i\|_{\Sigma_i}^2 \approx \lim_{i\to\infty} \sum_{j=0}^{i} \mathrm{Tr}\left(\mathcal{Z}_j\Sigma_j\right) \quad (37)$$

$$= \lim_{i\to\infty} \sum_{j=0}^{i} \mathrm{Tr}\left(\mathcal{Z}_j^{*/2}\Sigma_j\mathcal{Z}_j^{1/2}\right) = \lim_{i\to\infty} \sum_{j=0}^{i} \left\|\mathcal{Z}_j^{*/2}\Sigma_j\mathcal{Z}_j^{1/2}\right\|_*$$

$$\le \lim_{i\to\infty} \sum_{j=0}^{i} \|\mathcal{Z}_j^{1/2}\|_*^2 \cdot \|\Sigma_j\|_* \quad (38)$$

From (18) and (33), it can be verified that the error recursion (18) tends to a finite bias in the mean, i.e., $\lim_{i\to\infty}\mathbb{E}\widetilde{\boldsymbol{w}}_i < \infty$, because the coefficient matrix $\mathcal{B}_i$ is uniformly stable. Since each term on the right-hand side of (24) is uniformly bounded, let

$$\beta \triangleq \sup_{j\ge 0} \|\mathcal{Z}_j^{1/2}\|_*^2 < \infty \quad (39)$$

Then, from (35), (38), and (39), we get

$$\begin{aligned}
\lim_{i\to\infty} \mathbb{E}\|\widetilde{\boldsymbol{w}}_i\|_{\Sigma_i}^2 &\le \lim_{i\to\infty} \sum_{j=0}^{i} \beta\|\Sigma_j\|_* \\
&\le c^2\beta \lim_{i\to\infty} \sum_{j=0}^{i} [\rho(I_{NM} - \mathcal{M}\mathcal{R}_u)]^{2(i-j)}\|\Sigma_i\|_* \\
&= \frac{c^2\beta\|\Sigma_\infty\|_*}{1 - [\rho(I_{NM} - \mathcal{M}\mathcal{R}_u)]^2} \quad (40)
\end{aligned}$$

Therefore, the error recursion (18) is mean-square stable.

D. Steady-State Mean-Square Performance

We define the network MSD as

$$\mathrm{MSD} \triangleq \lim_{i\to\infty} \frac{1}{N} \sum_{k=1}^{N} \mathbb{E}\|\widetilde{\boldsymbol{w}}_{k,i}\|_2^2 \quad (41)$$

Selecting the steady-state weighting matrix as $\Sigma_\infty = I_{NM}/N$ and substituting it into (37), the network MSD is found to be

$$\mathrm{MSD}^{\text{adaptive}} \approx \lim_{i\to\infty} \frac{1}{N} \sum_{j=0}^{i} \mathrm{Tr}(\mathcal{B}_i \ldots \mathcal{B}_{j+1}\mathcal{Z}_j\mathcal{B}_{j+1}^* \ldots \mathcal{B}_i^*) \quad (42)$$

IV. Selecting the Combination Weights

The optimal combination matrix sequence $\{A_i\}$ can be obtained by solving:

$$\underset{\{A_i\}}{\text{minimize}} \quad \text{MSD}^{\text{adaptive}} \ \text{ in (42)} \tag{43}$$

subject to the topology constraints

$$A_i^{\mathsf{T}}\mathbb{1}_N = \mathbb{1}_N, \quad a_{lk} \geq 0, \quad a_{lk} = 0 \ \text{ if } l \notin \mathcal{N}_k \tag{44}$$

From (42), it is seen that the solution of problem (43) is *non-causal* since the choice of A_i depends on the past and future choices of $\{A_j\}$. To motivate a causal solution, we instead consider a sequential procedure that minimizes the *instantaneous* network MSD at each time i:

$$\underset{A_i}{\text{minimize}} \quad \frac{1}{N}\sum_{k=1}^{N}\mathbb{E}\|\widetilde{\boldsymbol{w}}_{k,i}\|_2^2 \tag{45}$$

subject to (44).

A. Minimizing Instantaneous MSD

From the combination step (7), we get

$$\begin{aligned}\mathbb{E}\|\widetilde{\boldsymbol{w}}_{k,i}\|_2^2 &= \mathbb{E}\left\|w_k^o - \sum_{l\in\mathcal{N}_k} a_{lk}(i)\boldsymbol{\psi}_{l,i}\right\|_2^2 \\ &= \sum_{l\in\mathcal{N}_k}\sum_{n\in\mathcal{N}_k} a_{lk}(i)a_{nk}(i)\mathbb{E}(w_k^o - \boldsymbol{\psi}_{n,i})^*(w_k^o - \boldsymbol{\psi}_{l,i})\end{aligned} \tag{46}$$

Let $W_{k,i}$ be an $N \times N$ matrix for each node k such that its (l,n)th entry is formed by the cross-covariances:

$$[W_{k,i}]_{ln} \triangleq \begin{cases}\mathbb{E}(w_k^o - \boldsymbol{\psi}_{n,i})^*(w_k^o - \boldsymbol{\psi}_{l,i}), & l,n \in \mathcal{N}_k \\ 0, & \text{otherwise}\end{cases} \tag{47}$$

It can be shown that $W_{k,i}$ is positive semi-definite. Let

$$a_{k,i} \triangleq \text{col}\{a_{1k}(i), a_{2k}(i), \ldots, a_{Nk}(i)\} \tag{48}$$

Then, the minimization problem (45) can be decoupled into N sub-problems, and each one of them can be formulated as

$$\boxed{\begin{array}{ll}\underset{\{a_{lk}(i);\, l\in\mathcal{N}_k\}}{\text{minimize}} & a_{k,i}^{\mathsf{T}} W_{k,i} a_{k,i} \\ \text{subject to} & a_{k,i}^{\mathsf{T}}\mathbb{1}_N = 1, \ \ a_{lk}(i) \geq 0, \\ & a_{lk}(i) = 0 \ \text{ if } \ l \notin \mathcal{N}_k\end{array}} \tag{49}$$

The solution is given by

$$\boxed{a_{k,i} = \frac{W_{k,i}^{-1}\mathbb{1}_N}{\mathbb{1}_N^{\mathsf{T}} W_{k,i}^{-1}\mathbb{1}_N}} \tag{50}$$

However, evaluating the off-diagonal entries of $W_{k,i}$ is generally non-trivial. Instead, we replace $W_{k,i}$ by its diagonal matrix and approximate (50) as:

$$\boxed{a_{lk}(i) \approx \frac{(\mathbb{E}\|w_k^o - \boldsymbol{\psi}_{l,i}\|_2^2)^{-1}}{\sum_{n\in\mathcal{N}_k}(\mathbb{E}\|w_k^o - \boldsymbol{\psi}_{n,i}\|_2^2)^{-1}}, \quad l \in \mathcal{N}_k} \tag{51}$$

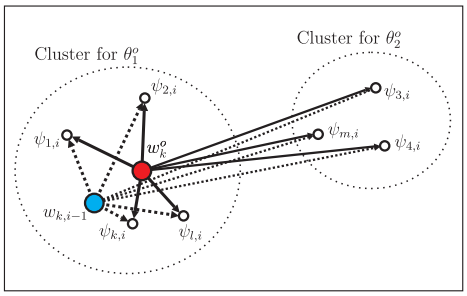


Fig. 1. Illustration of the adaptive rule (51) and its implementation (53) where w_k^o is assumed to be equal to θ_1^o.

This solution admits a physical interpretation: the combination weight assigned by node k to node l is inversely proportional to the (squared) displacement between the objective of node k and $\boldsymbol{\psi}_{l,i}$. Fig. 1 illustrates this construction.

Implementation of the combination rule (51) by node k requires knowledge of the objective w_k^o and the cross-covariance terms, which are generally not available beforehand. Nevertheless, we can employ an instantaneous approximation argument to address these two difficulties. We first use the previous estimate at node k, $w_{k,i-1}$, to replace w_k^o in (51). Then, we introduce the instantaneous metric:

$$\gamma_{lk}^2(i) \triangleq \|w_{k,i-1} - \psi_{l,i}\|_2^2 \tag{52}$$

and approximate the combination rule (51) by

$$\boxed{a_{lk}(i) \approx \frac{\gamma_{lk}^{-2}(i)}{\sum_{n\in\mathcal{N}_k}\gamma_{nk}^{-2}(i)}, \quad l \in \mathcal{N}_k} \tag{53}$$

We can further smooth the quantity $\gamma_{lk}^2(i)$ through a first-order filter, say,

$$\gamma_{lk}^2(i) = (1-\nu_k)\gamma_{lk}^2(i-1) + \nu_k\|w_{k,i-1} - \psi_{l,i}\|_2^2 \tag{54}$$

where $\{\nu_k\}$ are forgetting factors that are smaller than but close to one.

B. Summary of the Proposed Scheme

ATC Diffusion LMS with Adaptive Weights

Initialize $w_{k,-1}$ with random values and set $\gamma_{lk}^2(-1) = 0$ for all $k \in \{1,\ldots,N\}$ and $l \in \mathcal{N}_k$.

for $i \geq 0$ **do**

$$\begin{aligned}e_k(i) &= d_k(i) - u_{k,i}w_{k,i-1} \\ \psi_{k,i} &= w_{k,i-1} + \mu_k u_{k,i}^* e_k(i) \\ \gamma_{lk}^2(i) &= (1-\nu_k)\gamma_{lk}^2(i-1) + \nu_k\|w_{k,i-1} - \psi_{l,i}\|_2^2 \\ a_{lk}(i) &= \frac{\gamma_{lk}^{-2}(i)}{\sum_{n\in\mathcal{N}_k}\gamma_{nk}^{-2}(i)} \\ w_{k,i} &= \sum_{l\in\mathcal{N}_k} a_{lk}(i)\psi_{k,i}\end{aligned}$$

end for

C. Learning Behavior

It turns out that there are essentially two stages during the learning phase of the proposed diffusion clustering strategy. Significant cooperation among nodes occurs mainly during the

second stage. To see this, we reconsider the combination rule (53), namely,

$$a_{lk}(i) = \frac{\|w_{k,i-1} - \psi_{l,i}\|_2^{-2}}{\sum_{n\in\mathcal{N}_k} \|w_{k,i-1} - \psi_{n,i}\|_2^{-2}} \tag{55}$$

The combination weight assigned to node k is given by

$$\begin{aligned} a_{kk}(i) &= \frac{\|w_{k,i-1} - \psi_{k,i}\|_2^{-2}}{\sum_{n\in\mathcal{N}_k} \|w_{k,i-1} - \psi_{n,i}\|_2^{-2}} \\ &= \frac{\mu_k^{-2}|e_k(i)|^{-2}\|u_{k,i}\|_2^{-2}}{\sum_{n\in\mathcal{N}_k} \|w_{k,i-1} - \psi_{n,i}\|_2^{-2}} \end{aligned} \tag{56}$$

The contribution of node l to the new estimate $w_{k,i}$, relative to that of node k itself, can be assessed by the ratio $r_{lk}(i)$:

$$r_{lk}(i) \triangleq \frac{a_{lk}(i)}{a_{kk}(i)} = \frac{\mu_k^2|e_k(i)|^2\|u_{k,i}\|_2^2}{\|w_{k,i-1} - \psi_{l,i}\|_2^2} \tag{57}$$

where $|e_k(i)|^2 \sim O(\sigma_{v,k}^2)$ and $\|u_{k,i}\|_2^2 \sim O(\mathrm{Tr}(R_{u,k}))$. Let $\mathbb{B}(x;\delta)$ denote a 2-norm ball in the vector space $\mathbb{C}^{N\times 1}$ centered at x with radius $\delta > 0$, i.e., $\mathbb{B}(x;\delta) \triangleq \{y \in \mathbb{C}^{N\times 1}; \|y - x\|_2 < \delta\}$. We refer to this ball as the δ-near-field of x. Then, for a certain value of δ, say,

$$\delta = \sqrt{D\mu_k^2\sigma_{v,k}^2\mathrm{Tr}(R_{u,k})} \tag{58}$$

where $D \gg 1$, whenever $\psi_{l,i} \notin \mathbb{B}(w_{k,i-1};\delta)$, we get

$$r_{lk}(i) < \frac{\mu_k^2|e_k(i)|^2\|u_{k,i}\|_2^2}{\delta^2} = O\left(\frac{1}{D}\right) \ll 1 \tag{59}$$

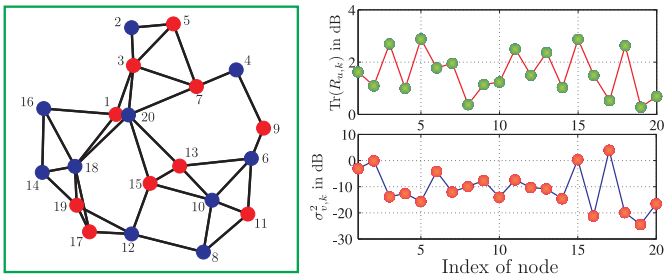
It means that before $\psi_{l,i}$ enters the δ-near-field $\mathbb{B}(w_{k,i-1};\delta)$, the relative ratio $r_{lk}(i)$ is negligible and thus the estimate $w_{k,i}$ is dominated by $\psi_{k,i}$ — the cooperation among nodes is insignificant during this stage. During this "far-field" stage, each node updates its estimate based mainly on its own data $\{d_k(i), u_{k,i}\}$. The update gradually drives the estimate towards $\mathbb{B}(w_k^o;\delta)$. Since only nodes sharing the same objective, say, θ_1^o, are able to converge to $\mathbb{B}(\theta_1^o;\delta)$ and cluster there, effective cooperation among nodes will only occur in a meaningful manner within the clusters. The closer nodes get into $\mathbb{B}(w_k^o;\delta)$, the larger the ratios $\{r_{lk}(i)\}$ will be. In this way, the ATC diffusion algorithm endows each node with the ability to differentiate its behavior with respect to its neighbors. The analysis suggests two conditions for convergence:

- The initialization of each $w_{k,-1}$ needs to be sufficiently away from the other initializations by at least δ, i.e., $\|w_{k,-1} - w_{l,-1}\|_2 > \delta$ for all k and l.
- The clustering vectors $\{\theta_m^o\}$ need to be sufficiently apart from each other by at least δ, i.e., $\|\theta_1^o - \theta_2^o\|_2 > \delta$.

where

$$\delta^2 = \max_k \; D\mu_k^2\sigma_{v,k}^2\mathrm{Tr}(R_{u,k}), \qquad D \gg 1 \tag{60}$$

The quantity δ reflects the discrimination ability of the network with respect to multiple clusters: if the objectives $\{\theta_m^o\}$ are too close to each other, then nodes will not be able to distinguish them from each other. Note from (60) that the value of δ is proportional to the step-sizes. In practice, for small step-sizes, selecting the initializations $\{w_{k,-1}\}$ randomly in space tends to be sufficient to guarantee convergence.



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

Fig. 2. Network topology with 20 nodes and related statistical profiles.

V. Simulation Results

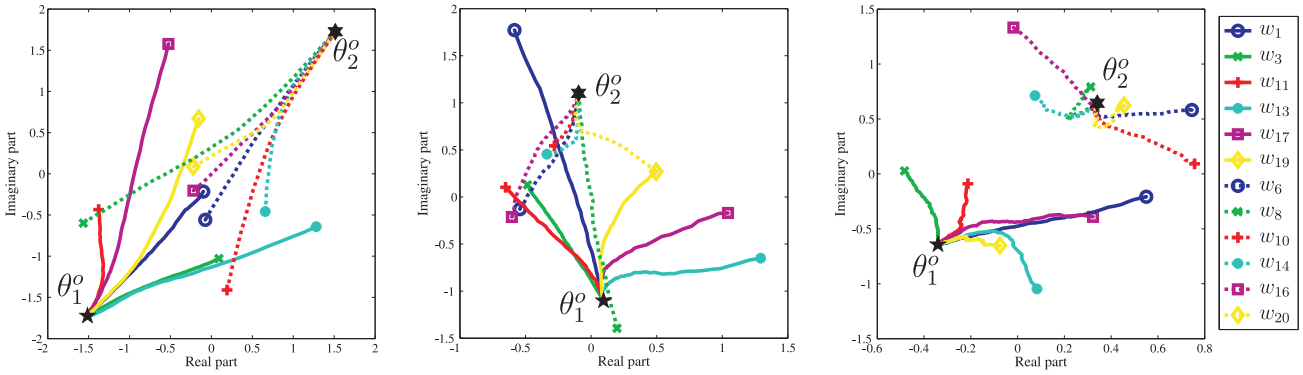
We simulate the ATC diffusion algorithm versus non-cooperative stand-alone LMS at each node over the connected network with $N = 20$ nodes shown in Fig. 2a. The two unknown parameters are $\theta_1^o = \theta$ and $\theta_2^o = -\theta$, where θ is of length $M = 3$ and is randomly generated. Nodes with odd index numbers (in red circles) are affected by data generated by θ_1^o while nodes with even index numbers (in blue circles) are affected by data generated by θ_2^o. The regression data are circular complex Gaussian with zero mean and covariance matrices $\{R_{u,k}\}$ that are randomly generated; their traces are shown in the upper part of Fig. 2b. The noise signals are also zero-mean circular complex Gaussian, whose variances, $\{\sigma_{v,k}^2\}$, are shown in the lower part of Fig. 2b. Both the regression data and the noise signals are temporally white and spatially independent. The step-size $\mu = 0.05$ and the forgetting factor $\nu = 0.1$ are uniform across the network. The initial values $\{w_{k,-1}\}$ are randomly generated.

Simulation results are shown in Figs. 3 and 4. We plot the trajectories for the values of $\{w_{k,i}\}$ in the complex plain in Figs. 3a–3c (the horizontal axis for the real part and the vertical axis for the imaginary part). These trajectories are averaged over 50 experiments with the same initial values for $\{w_{k,-1}\}$. They illustrate that the proposed diffusion algorithm guides each estimate $w_{k,i}$ towards its objective without being confused by irrelevant neighbors interested in different objectives.

In steady-state, we mapped the values of the estimates $\{w_{k,i}\}$ into the color of the circles in Fig. 4a. Edges are dropped if there weights are below a threshold value, say, $a_{lk}(i) < 0.05$. Compared to the topology in Fig. 2a, it can be seen that all nodes attain their desired objectives and cooperation only occurs among nodes sharing common objectives. The MSD learning curves are obtained by averaging over 50 experiments and are plotted in Fig. 4b. The theoretical result for the diffusion algorithm is obtained from (42), and the theoretical result for the non-cooperative LMS is obtained by [20, Ch. 16]:

$$\mathrm{MSD}^{\mathrm{noncooperation}} \approx \frac{1}{N}\sum_{k=1}^{N} \frac{\mu_k\sigma_{v,k}^2\mathrm{Tr}(R_{u,k})}{2} \tag{61}$$

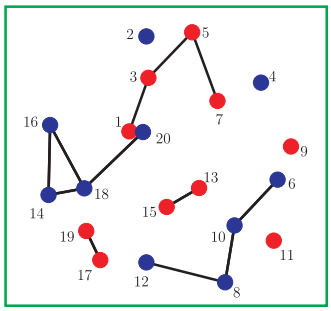
under Assumption 2. It can be seen that the proposed algorithm improves the network MSD performance by about 5 dB over



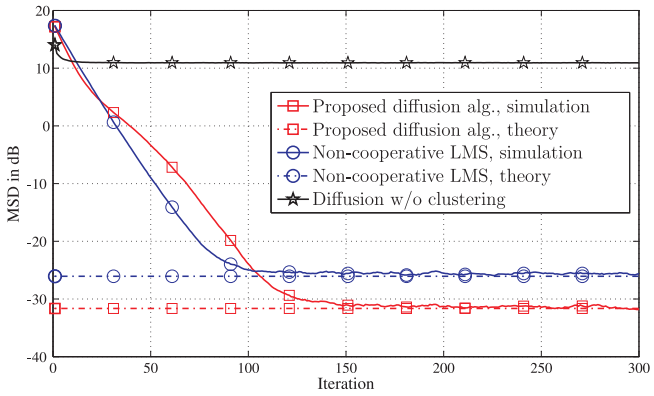
(a) First entry of $w_{k,i}$. (b) Second entry of $w_{k,i}$. (c) Third entry of $w_{k,i}$.

Fig. 3. Trajectories of the entries of $\{w_{k,i}\}$ over time.

the non-cooperative stand-alone LMS. In addition, we also plot the simulation results for the traditional diffusion algorithm (6)–(7) with uniform combination coefficients in Fig. 4b to illustrate the catastrophic result caused by uniform cooperation without discrimination.



(a) Steady-state network after clustering.



(b) Network MSD curves.

Fig. 4. Steady-state network MSD values and the corresponding MSD learning curves.

REFERENCES

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

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

[3] J. Tsitsiklis and M. Athans, "Convergence and asymptotic agreement in distributed decision problems," *IEEE Trans. Autom. Control*, vol. 29, no. 1, pp. 42–50, Jan. 1984.

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

[5] A. Nedic and D. P. Bertsekas, "Incremental subgradient methods for nondifferentiable optimization," *SIAM J. Optim.*, vol. 12, no. 1, pp. 109–138, 2001.

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

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

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

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

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

[11] J. Chen and A. H. Sayed, "Diffusion adaptation strategies for distributed optimization and learning over networks," *arXiv:1111.0034v2 [math.OC]*, Oct. 2011.

[12] J. Liu, M. Chu, and J. E. Reich, "Multitarget tracking in distributed sensor networks," *IEEE Signal Process. Mag.*, vol. 24, no. 3, pp. 36–46, May 2007.

[13] X. Zhang, "Adaptive control and reconfiguration of mobile wireless sensor networks for dynamic multi-target tracking," *IEEE Trans. Autom. Control*, vol. 56, no. 10, pp. 2429–2444, Oct. 2011.

[14] P. Bailis, R. Nagpal, and J. Werfel, "Positional communication and private information in honeybee foraging models," in *Proc. Int. Conf. Swarm Intell. (ANTS)*, Brussels, Belgium, Sept. 2010, pp. 263–274.

[15] D. T. Magill, "Optimal adaptive estimation of sampled stochastic processes," *IEEE Trans. Autom. Control*, vol. 10, no. 4, pp. 434–439, Oct. 1965.

[16] I. Francis and S. Chatterjee, "Classification and estimation of several multiple regressions," *The Annals of Statistics*, vol. 2, no. 3, pp. 558–561, 1974.

[17] X.-R. Li and Y. Bar-Shalom, "Multiple-model estimation with variable structure," *IEEE Trans. Autom. Control*, vol. 41, no. 4, pp. 478–493, Apr. 1996.

[18] X.-R. Li, "Multiple-model estimation with variable structure – part ii: Model-set adaptation," *IEEE Trans. Autom. Control*, vol. 45, no. 11, pp. 2047–2060, Nov. 2000.

[19] V. Cherkassky and Y. Ma, "Multiple model regression estimation," *IEEE Trans. Neural Netw.*, vol. 16, no. 4, pp. 785–798, July 2005.

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

[21] X. Zhao, S.-Y. Tu, and A. H. Sayed, "Diffusion adaptation over netowrks under imperfect information exchange and non-stationary data," *IEEE Trans. Signal Process.*, accepted for publication.

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

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