A DIFFUSION LMS STRATEGY FOR PARAMETER ESTIMATION IN NOISY REGRESSOR APPLICATIONS

Reza Abdolee, Benoit Champagne[*] *Ali H. Sayed*[†]

McGill University
Dept. of Elec. and Comp. Engineering
Montreal, QC, H3A 2A7, Canada

University of California
Dept. of Electrical Engineering
Los Angeles, CA, 90095, USA

ABSTRACT

We study distributed least-mean square (LMS) estimation problems over adaptive networks, where nodes cooperatively work to estimate and track common parameters of an unknown system. We consider a scenario where the input and output response signals of the unknown system are both contaminated by measurement noise. In this case, if standard distributed estimation is performed without considering the effect of regression noise, then the resulting parameter estimates will be biased. To resolve this problem, we propose a distributed LMS algorithm that achieves asymptotically unbiased estimates via diffusion adaptation. We analyze the performance of the proposed algorithm and provide computer experiments to illustrate its behavior.

Index Terms— distributed estimation, diffusion adaptation, cooperative processing, bias-compensated LMS, noisy regressor.

1 Introduction

In adaptive parameter estimation, when the regression data used by an LMS filter are contaminated by noise, the estimates generated by the filter will be biased and unreliable [1]. Bias removal and compensation for *stand-alone* LMS filters have been addressed before in the literature [2–8]. In this work, we study the problem in the context of *distributed* processing where a network of N interconnected nodes work cooperatively to estimate an unknown parameter vector w^o.

To mitigate the effect of measurement noise and regressor perturbation, a distributed total-least-squares (DTLS) algorithm, based on semidefinite relaxation and convex semidefinite programming, was proposed in [9]. This algorithm requires the eigendecomposition of the augmented covariance matrix in every iteration at each node, and is therefore mainly suitable for applications involving nodes with powerful processing abilities. A similar problem was studied in [10] in the context of recursive least-squares (RLS) estimation over networks. The resulting RLS based algorithms are attractive because of their fast convergence rate. However, their high computational complexity and numerical instability are generally a hindrance. For this reason, we are motivated to examine the ability of LMS diffusion strategies to correct for the bias through cooperation. Specifically, we propose a distributed LMS strategy based on a bias-elimination technique that exploits both the spatial diversity of the data and the regressor noise variance information to attain an unbiased estimate via an adaptive diffusion process.

Notation: We use boldface notation to represent random variables, and normal font to represent deterministic quantities. For complex vectors and matrices, $(\cdot)^*$ denotes complex conjugate transposition. I_M denotes the identity matrix of size $M \times M$.

2 Stand-Alone LMS Filtering

We explain the concept of the proposed algorithm using a system identification set-up, illustrated in Fig. 1. As shown in the figure, the adaptive filter has access to noisy perturbations of the regression data and to noisy measurements of the output of a tapped-delay-line with impulse response vector w^o. We denote the $1 \times M$ regression vector by $\mathbf{u}_i$ and its noisy version by $\mathbf{z}_i = \mathbf{u}_i + \mathbf{n}_i$, where i denotes the time index. Likewise, we denote the uncorrupted scaler output of the system by $\mathbf{y}(i) = \mathbf{u}_i w^o$ and its noisy version by $\mathbf{d}(i) = \mathbf{y}(i) + \mathbf{v}(i)$. All processes are assumed to be wide-sense stationary and have zero means. Moreover, the noises $\{\mathbf{n}_i, \mathbf{v}(i)\}$ are assumed to be temporally white and independent of each other. Under these conditions, when an LMS filter is used to estimate w^o, in steady-state, as $i \to \infty$, the mean value of the filter estimate does not tend to w^o but is biased and its value is away from w^o by an amount that is

[*]The work of R. Abdolee and B. Champagne was supported in part by the Fonds Québécois de la recherche sur la nature et les technologies (FQRNT) and the Natural Sciences and Engineering Research Council (NSERC) of Canada.

[†]The work of A. H. Sayed was supported in part by NSF grants CCF-1011918 and CCF-0942936.

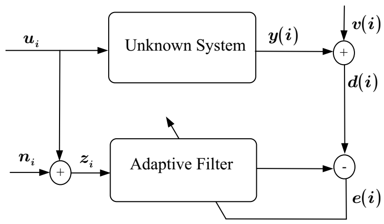


Fig. 1: An adaptive system identification configuration showing noisy perturbations of both the regression data and the reference data.

approximately equal to $R_u^{-1}R_n w^o$, where $R_u = \mathrm{E}\mathbf{u}_i^*\mathbf{u}_i$, $R_n = \mathrm{E}\mathbf{n}_i^*\mathbf{n}_i = \sigma_n^2 I_M$. This is because, on average, the estimate of an LMS filter converges towards

$$w_z^o = (R_u + \sigma_n^2 I_M)^{-1} R_{du} \tag{1}$$

where $R_{du} = \mathrm{E}\mathbf{u}_i^*\mathbf{d}(i)$. For large signal-to-noise ratio, it holds that

$$(R_u + \sigma_n^2 I_M)^{-1} \approx R_u^{-1} - \sigma_n^2 R_u^{-2} \tag{2}$$

Substituting (2) in (1) leads to

$$w_z^o \approx w^o - \sigma_n^2 R_u^{-1} w^o \tag{3}$$

where $w^o = R_u^{-1}R_{du}$ is the unbiased solution in the absence of regression noise [11]. One way to remove the bias is to consider the following modified mean-square-error cost function:

$$J(w) = \mathrm{E}|\mathbf{d}(i) - \mathbf{z}_i w|^2 - \sigma_n^2\|w\|^2 \tag{4}$$

where $\|\cdot\|$ and $\mathrm{E}[\cdot]$ denote the Euclidean norm and expectation operator, respectively. A stochastic gradient algorithm for solving (4) takes the following form:

$$\mathbf{e}(i) = \mathbf{d}(i) - \mathbf{z}_i\mathbf{w}_{i-1} \tag{5}$$

$$\mathbf{w}_i = \mathbf{w}_{i-1} + \mu(\mathbf{z}_i^*\mathbf{e}(i) + \sigma_n^2\mathbf{w}_{i-1}) \tag{6}$$

where $\mu > 0$ is the step-size. In what follows, we use the above strategy to develop a bias-compensated diffusion LMS algorithm for adaptation over networks.

3 Bias-Compensated Diffusion LMS

Consider a connected network of N nodes, observing data $\{\mathbf{d}_k(i), \mathbf{z}_{k,i}\}$ that satisfy the model:

$$\mathbf{d}_k(i) = \mathbf{u}_{k,i}w^o + \mathbf{v}_k(i) \tag{7}$$

$$\mathbf{z}_{k,i} = \mathbf{u}_{k,i} + \mathbf{n}_{k,i} \tag{8}$$

where i and k are time and space indices, respectively, $\mathbf{u}_{k,i} \in \mathbb{C}^{1\times M}$ is the regression data, $w^o \in \mathbb{C}^{M\times 1}$ represents the unknown parameter vector to be estimated, and $\mathbf{n}_{k,i} \in \mathbb{C}^{1\times M}$ and $\mathbf{v}_k(i) \in \mathbb{C}$ are input and output measurement noise, respectively. The following conditions are assumed for this model. The regression data $\{\mathbf{u}_{k,i}\}$ are zero mean, i.i.d over time and independent over space with covariance matrices $R_{u,k} = \mathrm{E}\mathbf{u}_{k,i}^*\mathbf{u}_{k,i} > 0$. The noise process $\mathbf{v}_k(i)$ is zero-mean and temporally white and independent over space with variance $\sigma_{v,k}^2$. This noise is also independent of the regressors $\{\mathbf{u}_{m,j}\}$ for all k, m, i, j. The elements of the regressor noise $\mathbf{n}_{k,i}$ are zero-mean jointly Gaussian random variables independent in time and space with diagonal covariance matrix $R_{n,k} = \sigma_{n,k}^2 I_M$. The regressor noise is also independent from $\mathbf{v}_m(j)$, $\mathbf{d}_m(j)$ and $\mathbf{u}_{m,j}$ for all k, m, i, j.

To determine an unbiased steady-state estimate for w^o, we propose a bias-compensated diffusion LMS algorithm in which each node k exchanges information with its neighbors $\ell \in \mathcal{N}_k$, where $\mathcal{N}_k = \{\nu_{k,1}, \nu_{k,2}, \dots \nu_{k,n_k}\}$ is the neighborhood of node k, $\nu_{k,\ell} \in \mathbb{Z}^+$ is the neighbor index, and $n_k = |\mathcal{N}_k|$ is the number of neighbors of node k. We introduce combination matrices C and $A \in \mathbb{R}^{N\times N}$, with non-negative entries $c_{\ell,k}$ and $a_{\ell,k}$ that satisfy $c_{k,\ell} = a_{k,\ell} = 0$, if $\ell \notin \mathcal{N}_k$, and $C\mathbb{1}_N = A^T\mathbb{1}_N = \mathbb{1}_N$ where $\mathbb{1}_N \in \mathbb{R}^{N\times 1}$ is a vector with unit entries. We then seek a distributed solution that minimizes the following global cost across the network:

$$J_{\text{glob}}(w) = \sum_{k=1}^{N} \mathrm{E}|\mathbf{d}_k(i) - \mathbf{z}_{k,i}w|^2 - \sigma_{n,k}^2\|w\|^2 \tag{9}$$

Following the arguments of [12, 13], we can derive the following diffusion LMS algorithm for distributed computation of w^o, where $\mu_k > 0$ is the algorithm step-size at node k.

Algorithm 1 Bias-compensated diffusion LMS

$$\boldsymbol{\psi}_{k,i} = \mathbf{w}_{k,i-1} + \mu_k \sum_{\ell\in\mathcal{N}_k} c_{\ell,k}\left[\mathbf{z}_{\ell,i}^*(\mathbf{d}_\ell(i) - \mathbf{z}_{\ell,i}\mathbf{w}_{k,i-1}) + \sigma_{n,\ell}^2\mathbf{w}_{k,i-1}\right]$$

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

In Algorithm 1, the first and the second update equations are called the adaptation and combination steps, respectively. In the adaptation step, at every iteration i, every node k first update its existing estimate $\mathbf{w}_{k,i-1}$ to an intermediate value $\boldsymbol{\psi}_{k,i}$ using data $\{\mathbf{d}_\ell(i), \mathbf{u}_{\ell,i}, \sigma_{n,\ell}^2\}$ from its neighbors. Subsequently, in the combination step, the intermediate estimates from across the neighborhood of node k are combined to yield $\mathbf{w}_{k,i}$.

There are several ways by which the combination matrices C and A can be selected [12, 13]. In this article, we use Metropolis weights for C and assume the entries of A are computed as follows:

$$a_{\ell,k} = \frac{1/\sigma_{n,\ell}^2}{\sum_{\ell\in\mathcal{N}_k} 1/\sigma_{n,\ell}^2} \tag{10}$$

By using this rule, node k gives more weight to neighbors with smaller regression noise variance. The stated form of Algorithm 1 requires knowledge of the regression noise variances, $\sigma^2_{n,\ell}$. These variances are generally unknown beforehand. In future work, we will examine procedures to estimate $\sigma^2_{n,\ell}$ in a manner similar to techniques described in, for example, [13,14] for the estimation of noise link variances.

4 Performance Analysis

In this section, we summarize the main analysis results. In particular, we show that the algorithm is unbiased, and stable in the mean, and derive closed-form expressions to predict the steady-state MSD and EMSE over the network. The derivation are omitted for brevity.

4.1 Convergence in the Mean

We begin by defining the local weight error vectors $\tilde{\mathbf{w}}_{k,i} = w^o - \mathbf{w}_{k,i}$ and $\tilde{\boldsymbol{\psi}}_{k,i} = w^o - \boldsymbol{\psi}_{k,i}$. Subtracting w^o from the both sides of the first update equation of Algorithm 1, leads to

$$\tilde{\boldsymbol{\psi}}_{k,i} = \tilde{\mathbf{w}}_{k,i-1} - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell,k}[\mathbf{z}^*_{\ell,i}(\mathbf{d}_\ell(i) - \mathbf{z}_{\ell,i}\mathbf{w}_{k,i-1}) + \sigma^2_{n,\ell}\mathbf{w}_{k,i-1}] \tag{11}$$

By using $\mathbf{v}_\ell(i) = \mathbf{d}_\ell(i) - \mathbf{u}_{\ell,i} w^o$ in (11), we obtain

$$\tilde{\boldsymbol{\psi}}_{k,i} = \tilde{\mathbf{w}}_{k,i-1} - \mu_k \sum_{\ell \in \mathcal{N}_k} c_{\ell,k}\Big\{\mathbf{z}^*_{\ell,i}\mathbf{v}_\ell(i) - (\mathbf{z}^*_{\ell,i}\mathbf{n}_{\ell,i} - \sigma^2_{n,\ell} I_M) w^o + (\mathbf{z}^*_{\ell,i}\mathbf{z}_{\ell,i} - \sigma^2_{n,\ell} I_M)\tilde{\mathbf{w}}_{k,i-1}\Big\} \tag{12}$$

By subtracting w^o from the both sides of the second update equation in Algorithm 1, we get

$$\tilde{\mathbf{w}}_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell,k} \tilde{\boldsymbol{\psi}}_{\ell,i} \tag{13}$$

Let us introduce:

$$\boldsymbol{\mathcal{D}}_i = \text{diag}\Big\{ \sum_{\ell=1}^{N} c_{\ell,j}\,(\mathbf{z}^*_{\ell,i}\mathbf{z}_{\ell,i} - \sigma^2_{n,\ell} I_M), j = 1, \cdots, N \Big\} \tag{14}$$

$$\mathcal{M} = \text{diag}\Big\{ \mu_1 I_M, \cdots, \mu_N I_M \Big\} \tag{15}$$

$$\boldsymbol{g}_i = \mathcal{C}^T \text{col}\{\mathbf{z}^*_{1,i}\mathbf{v}_1(i), \cdots, \mathbf{z}^*_{N,i}\mathbf{v}_N(i)\} \tag{16}$$

$$\boldsymbol{\mathcal{P}}_i = \text{diag}\Big\{ \sum_{\ell=1}^{N} c_{\ell,j}\,(\mathbf{z}^*_{\ell,i}\mathbf{n}_{\ell,i} - \sigma^2_{n,\ell} I_M), j = 1, \cdots, N \Big\} \tag{17}$$

and define the network error vectors by stacking up all the local error vectors on top of each other, i.e.,

$$\tilde{\boldsymbol{\psi}}_i = \text{col}\{\tilde{\boldsymbol{\psi}}_{1,i}, \tilde{\boldsymbol{\psi}}_{2,i}, \ldots, \tilde{\boldsymbol{\psi}}_{N,i}\} \tag{18}$$

$$\tilde{\mathbf{w}}_i = \text{col}\{\tilde{\mathbf{w}}_{1,i}, \tilde{\mathbf{w}}_{2,i}, \ldots, \tilde{\mathbf{w}}_{N,i}\} \tag{19}$$

In addition, we introduce the extended weighting matrices $\mathcal{A} = A \otimes I_M$, $\mathcal{C} = C \otimes I_M$, where $\otimes$ is the Kronecker product operation. We now can use (12) and (13) to write:

$$\begin{aligned} \tilde{\boldsymbol{\psi}}_i &= \tilde{\mathbf{w}}_{i-1} - \mathcal{M}(\boldsymbol{g}_i - \boldsymbol{\mathcal{P}}_i w^o + \boldsymbol{\mathcal{D}}_i \tilde{\mathbf{w}}_{i-1}) \\ \tilde{\mathbf{w}}_i &= \mathcal{A}^T \tilde{\boldsymbol{\psi}}_i \end{aligned} \tag{20}$$

and, hence,

$$\tilde{\mathbf{w}}_i = \mathcal{A}^T(I - \mathcal{M}\boldsymbol{\mathcal{D}}_i)\tilde{\mathbf{w}}_{i-1} - \mathcal{A}^T\mathcal{M}\boldsymbol{g}_i + \mathcal{A}^T\mathcal{M}\boldsymbol{\mathcal{P}}_i w^o \tag{21}$$

By taking the expectation of both sides, we obtain:

$$\text{E}[\tilde{\mathbf{w}}_i] = \mathcal{A}^T(I - \mathcal{M}\mathcal{D})\text{E}[\tilde{\mathbf{w}}_{i-1}] \tag{22}$$

where we used the fact that $\text{E}[\mathcal{A}^T\mathcal{M}\boldsymbol{g}_i] = 0$ because $\text{E}[\mathbf{v}_k(i)] = 0$ and $\mathbf{v}_k(i)$ is independent of $\mathbf{z}_{m,j}$ for all k, m and i, j. Moreover, in (21), $\text{E}[\boldsymbol{\mathcal{P}}_i] = 0$ because $\text{E}[\mathbf{z}^*_{k,i}\mathbf{n}_{k,i}] = \sigma^2_{n,k} I_M$ for $k \in \{1, \cdots, N\}$. In (22), $\mathcal{D} = \text{E}[\boldsymbol{\mathcal{D}}_i]$ is given by:

$$\mathcal{D} = \text{diag}\Big\{ \sum_{\ell=1}^{N} c_{\ell,1} R_{u,\ell}, \ldots, \sum_{\ell=1}^{N} c_{\ell,N} R_{u,\ell} \Big\} \tag{23}$$

The average weight error vector, $\text{E}[\tilde{\mathbf{w}}_i]$, in (22) vanishes as $i \longrightarrow \infty$ if the step-sizes are chosen as in the theorem below [13].

Theorem 1. *Assume Algorithm 1 runs over a network with space-time data satisfying (7) and (8). Then, if the step-sizes are chosen to satisfy:*

$$0 < \mu_k < \frac{2}{\lambda_{\max}(\sum_{\ell \in \mathcal{N}_k} c_{\ell,k} R_{u,\ell})} \tag{24}$$

the algorithm is asymptotically unbiased. In (24), $\lambda_{\max}(\cdot)$ denotes the largest eigenvalue of its matrix argument.

4.2 Mean-Square Convergence

By using the energy conservation approach of [11], we can establish that for sufficiently small step-sizes (so that higher-order powers of the step-sizes can be ignored) the following relation holds [13]:

$$\begin{aligned} \text{E}\|\tilde{\mathbf{w}}_i\|^2_\Sigma =& \text{E}\|\tilde{\mathbf{w}}_{i-1}\|^2_{\Sigma'} + \text{Tr}[\Sigma\mathcal{A}^T\mathcal{M}\mathcal{G}\mathcal{M}\mathcal{A}] \\ &+ \text{Tr}[\Sigma\mathcal{A}^T\mathcal{M}\mathcal{P}\mathcal{M}\mathcal{A}] \end{aligned} \tag{25}$$

where $\Sigma \geq 0$ is a weighting matrix that we are free to choose, and

$$\Sigma' = \text{E}(I_{MN} - \mathcal{M}\boldsymbol{\mathcal{D}}_i)^*\mathcal{A}\Sigma\mathcal{A}^T(I_{MN} - \mathcal{M}\boldsymbol{\mathcal{D}}_i) \tag{26}$$

In equation (25), $\mathcal{G} = \text{E}[\boldsymbol{g}^*_i\boldsymbol{g}_i]$ is computed as:

$$\mathcal{G} = \mathcal{C}^T\text{diag}\big\{\sigma^2_{v,1}(R_{u,1} + \sigma^2_{n,1} I_{MN}), \cdots, \sigma^2_{v,N}(R_{u,N} + \sigma^2_{n,N} I_{MN})\big\}\mathcal{C} \tag{27}$$

and $\mathcal{P} = \text{E}[\boldsymbol{\mathcal{P}}^*_i w^o w^{o*} \boldsymbol{\mathcal{P}}_i]$ where its $[k, j]^{th}$ block is given by

$$\mathcal{P}[k, j] = \sum_{\ell=1}^{N} c_{\ell,k} c_{\ell,j}\, \sigma^2_{n,\ell} \|w^o\|^2 (R_{u,\ell} + \beta\sigma^2_{n,\ell} I_M) \tag{28}$$

where $\beta = 1$ for complex-valued regression noise and $\beta = 2$ for real-valued regression noise. Let $\sigma = \text{bvec}(\Sigma)$ and $\sigma' = \text{bvec}(\Sigma')$, where the block vectorization operation $\text{bvec}(\Sigma)$ first vectorizes each block entry of Σ and then places all the resulting vectors on top of each other [15]. Then, the variance relation in (25) becomes:

$$\text{E}\|\tilde{\mathbf{w}}_i\|^2_\sigma = \text{E}\|\tilde{\mathbf{w}}_{i-1}\|^2_{\sigma'} + \big[\text{bvec}(\mathcal{A}^T\mathcal{M}\mathcal{G}\mathcal{M}\mathcal{A} + \mathcal{A}^T\mathcal{M}\mathcal{P}\mathcal{M}\mathcal{A})\big]^T\sigma \quad (29)$$

where we are using the notation $\|x\|^2_\sigma$ to also refer to $\|x\|^2_\Sigma$. It can be verified that the diffusion filter is mean-square stable for sufficiently small step-sizes. Therefore, letting $i \to \infty$, expression (29) leads to:

$$\lim_{i\to\infty} \text{E}\|\tilde{\mathbf{w}}_i\|^2_{(I_L - F)\sigma} = [\text{bvec}(\mathcal{A}^T\mathcal{M}\mathcal{G}\mathcal{M}\mathcal{A} + \mathcal{A}^T\mathcal{M}\mathcal{P}\mathcal{M}\mathcal{A})]^T\sigma \quad (30)$$

where $L = (MN)^2$ and

$$F \approx \Big(I_L - I_{MN} \odot (\mathcal{D}^*\mathcal{M}) - (\mathcal{D}^T\mathcal{M}) \odot I_{MN}\Big)(\mathcal{A} \odot \mathcal{A}) \quad (31)$$

where $\odot$ represents the *block Kronecker product* [15]. Let:

$$\eta_k = \lim_{i\to\infty} \text{E}\|w^o - \mathbf{w}_{k,i}\|^2 = \lim_{i\to\infty} \text{E}\|\tilde{\mathbf{w}}_{k,i}\|^2_{I_M} \quad (32)$$

$$\zeta_k = \lim_{i\to\infty} \text{E}|\mathbf{u}_{k,i}\tilde{\mathbf{w}}_{k,i}|^2 = \lim_{i\to\infty} \text{E}\|\tilde{\mathbf{w}}_{k,i}\|^2_{R_{u,k}} \quad (33)$$

These parameters can be retrieved from the global error vector as:

$$\eta_k = \lim_{i\to\infty} \text{E}\|\tilde{\mathbf{w}}_i\|^2_{\{\text{diag}(e_k)\otimes I_M\}} \quad (34)$$

$$\zeta_k = \lim_{i\to\infty} \text{E}\|\tilde{\mathbf{w}}_i\|^2_{\{\text{diag}(e_k)\otimes R_{u,k}\}} \quad (35)$$

where $\{e_k\}_{k=1}^N$ are canonical basis vectors of dimension N. By using these relations and (30), we can obtain the MSD and EMSE at node k:

$$\eta_k = \big[\text{bvec}(\mathcal{A}^T\mathcal{M}\mathcal{G}\mathcal{M}\mathcal{A} + \mathcal{A}^T\mathcal{M}\mathcal{P}\mathcal{M}\mathcal{A})\big]^T\sigma_{\text{msd}_k} \quad (36)$$

$$\zeta_k = \big[\text{bvec}(\mathcal{A}^T\mathcal{M}\mathcal{G}\mathcal{M}\mathcal{A} + \mathcal{A}^T\mathcal{M}\mathcal{P}\mathcal{M}\mathcal{A})\big]^T\sigma_{\text{emse}_k} \quad (37)$$

where

$$\sigma_{\text{msd}_k} = (I_L - F)^{-1}\text{bvec}(\text{diag}(e_k) \otimes I_M) \quad (38)$$

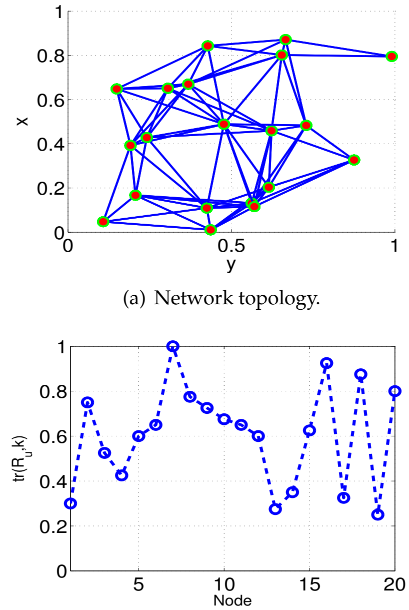
$$\sigma_{\text{emse}_k} = (I_L - F)^{-1}\text{bvec}(\text{diag}(e_k) \otimes R_{u,k}) \quad (39)$$

The network MSD and EMSE are defined as the average of the MSD and EMSE values over the network, i.e.,

$$\eta_{\text{net}} = \frac{1}{N}\sum_{k=1}^{N}\eta_k, \quad \zeta_{\text{net}} = \frac{1}{N}\sum_{k=1}^{N}\zeta_k \quad (40)$$

5 Simulation Results

We consider a connected network with $N = 20$ nodes that are placed randomly on a unit square area where the maximum communication range of each node is less than 0.4 unit length. The network aims to cooperatively estimate the unknown system parameter $w^o = [1, 1]^T/\sqrt{2}$. We choose the same step sizes $\mu_k = 0.05$ for all nodes, and initialize each local filter with zero. We compute the network weighting matrix C according to the Metropolis rule [15], and the combination matrix A as (10). In the results, we use C_{met} and A_{rel} to refer to these matrices. To show the noise profile over the network, we use the contours of the noise variances in three dimensional space in which the colorbar is used to decode the magnitude of the noise variance.



(a) Network topology.

(b) Regressor power, $\text{Tr}(R_{u,k})$.

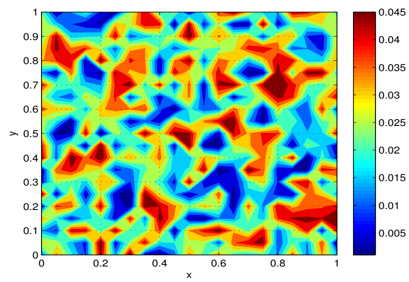
Fig. 2: Network topology and energy profile.

The network topology and energy profile are shown in Fig. 2. The contours of noise variances $\sigma^2_{n,k}$ and $\sigma^2_{v,k}$ over space in x and y dimensions are shown in Fig. 3. In Figs. 4(a) and 4(b), we demonstrate the network performance in terms of steady-state MSD and EMSE for different weighting and combination matrices. Note that the choice $A = I_{MN}$ and $C = I_{MN}$ corresponds to the non-cooperative scenario, where nodes work independently from one another. As the results indicate, the cooperative network scenario, with C_{met} and A_{rel}, has a 12dB superior performance over the non-cooperative case. Moreover, the simulation results match well with theory.

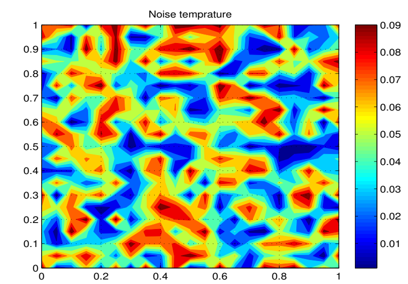
We also observe that the performance discrepancies between the nodes in terms of MSD is less than 0.5 dB for cooperative scenarios whereas it is more than 5dB in the non-cooperative scenario.

6 Conclusion

We developed a distributed LMS algorithm for parameter estimation in networking applications where the regression data are corrupted by noise. The proposed al-

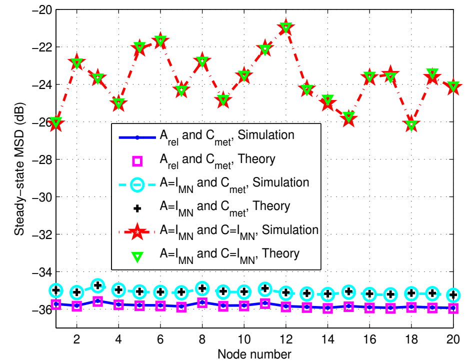


(a) Regressor noise variance $\sigma^2_{n,k}$.

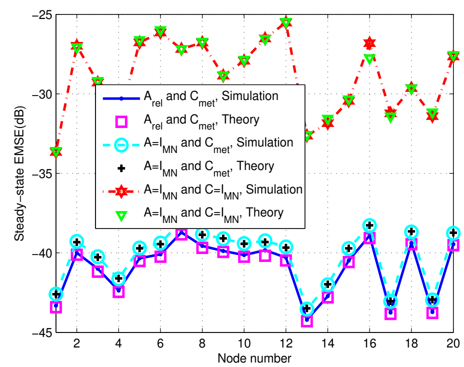


(b) Noise variance $\sigma^2_{v,k}$.

Fig. 3: Network noise profile.



(a) Network steady-state MSD.



(b) Network steady-state EMSE.

Fig. 4: Network performance comparison for different combination matrices.

gorithm exploits the spatial diversity of the data as well as regressor noise variance to estimate the unknown parameters and to keep tracking the parameters changes over time. The algorithm was verified to be asymptotically unbiased for sufficiently small step-sizes.

7 References

[1] D. Feng, Z. Bao, and L. Jiao, "Total least mean squares algorithm," *IEEE Trans. Signal Process.*, vol. 46, pp. 2122–2130, Aug. 1998.

[2] S. Sagara and K. Wada, "On-line modified least-squares parameter estimation of linear discrete dynamic systems," *Int. J. of Control*, pp. 329–343, Mar. 1977.

[3] W. Zheng and C. Feng, "Unbiased parameter estimation of linear systems in the presence of input and output noise," *Int. J. of Adaptive Control and Sig. Process.*, pp. 231–251, Mar. 1989.

[4] C. Davila, "An efficient recursive total least squares algorithm for FIR adaptive filtering," *IEEE Trans. Signal Process.*, vol. 42, pp. 268–280, Feb. 1994.

[5] H. So, "Modified LMS algorithm for unbiased impulse response estimation in nonstationary noise," *Electronics Letters*, pp. 791–792, Oct. 1999.

[6] D. Feng, Z. Bao, and X. Zhang, "Modified RLS algorithm for unbiased estimation of FIR system with input and output noise," *Electronics Letters*, pp. 273–274, Mar. 2000.

[7] L. Jia, M. Ikenoue, C. Jin, and K. Wada, "On bias compensated least squares method for noisy input-output system identification," in Proc. *IEEE Conf. on Decision and Control*, Aug. 2001, pp. 3332–3337.

[8] S. Jo and S. Kim, "Consistent normalized least mean square filtering with noisy data matrix," *IEEE Trans. Signal Process.*, vol. 53, pp. 2112–2123, July 2005.

[9] A. Bertrand and M. Moonen, "Consensus-based distributed total least squares estimation in ad hoc wireless sensor networks," *IEEE Trans. Signal Process.*, pp. 2320–2330, May 2011.

[10] A. Bertrand, M. Moonen, and A. H. Sayed, "Diffusion bias-compensated RLS estimation over adaptive networks," *IEEE Trans. Signal Process.*, pp. 5212–5224, Nov. 2011.

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

[12] F. S. Cattivelli and A. H. Sayed, "Diffusion LMS strategies for distributed estimation," *IEEE Trans. Signal Process.*, vol. 58, pp. 1035–1048, Mar 2010.

[13] A. Sayed, "Diffusion adaptation over networks," available online at http://arxiv.org/abs/1205.4220 as manuscript *arXiv:1205.4220v1 [cs.MA]*, May 2012.

[14] X. Zhao, S. Y. Tu, and A. H. Sayed, "Diffusion adaptation over networks under imperfect information exchange and non-stationary data," to appear in *IEEE Trans. on Signal Processing, 2012.* Also available on arXiv:1112.6212v3, Dec. 2011.

[15] C. G. Lopes and A. H. Sayed, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 56, pp. 3122–3136, July 2008.