Forty-Fifth Annual Allerton Conference
Allerton House, UIUC, Illinois, USA
September 26-28, 2007

Consensus-Based Distributed Least-Mean Square Algorithm Using Wireless Ad Hoc Networks†

Gonzalo Mateos, Ioannis D. Schizas and Georgios B. Giannakis
Dept. of ECE, University of Minnesota
200 Union Str. SE, Minneapolis, MN 55455, USA
Emails: {mate0058,schiz001,georgios}@umn.edu

***Abstract*— We deal with online estimation and tracking of (non)stationary signals using ad hoc wireless sensor networks (WSNs). A fully *distributed* least-mean square (D-LMS) type of algorithm is presented, which offers simplicity and flexibility whilst solely requiring single-hop communications among sensors. The algorithm entails the minimization of a pertinent cost function by resorting to: (i) the alternating-direction method of multipliers so as to gain the desired degree of parallelization and, (ii) stochastic approximation ideas to cope with the unavailability of statistical information. First-order convergence analysis is provided whereas mean-square sense (mss) convergence is corroborated via simulations, even in the presence of additive reception noise.**

I. INTRODUCTION

Driven by a wide span of foreseen applications, distributed estimation of signals based on observations collected by spatially dispersed sensors has attracted much attention recently. Specifically, ad hoc WSNs based on power efficient single hop communications raise several exciting challenges when targeted for signal processing tasks. The lack of hierarchy and purely decentralized nature of processing dictates that local sensor estimates should eventually consent to the desired quantity and exhaustively exploit the spatial dimension to maximize performance. For prior works on estimation in ad hoc WSNs, see e.g., [8] and references therein.

The incremental LMS algorithm in [7] became the first exponent among the online adaptive decentralized estimation schemes, which incorporate new available information in real-time and can accommodate time variations in the signal statistics. Although this approach requires limited amount of communications, its inherent requirement of a Hamiltonian cycle through which the estimate is circulated poses a serious obstacle in practical deployments. In the eventuality of a sensor failure, determination of a new cycle is a well-known NP-complete problem in graph theory [6], and thus infeasible for medium to large networks. Avoiding the need of such a cycle and further exploiting the exchange of information among neighbors to yield improved local estimates, the diffusion LMS [5], [7] offers an improved alternative with increased communication cost and a somewhat heuristic derivation.

Here we develop a consensus-based distributed LMS algorithm for use in general ad hoc WSNs with noisy links, whose simplicity is well-matched to the power and communication resource scarcity characterizing these networks. Our algorithm is derived from a well-posed optimization problem defining the desired estimator (Section II). Building on [8] we reformulate the original problem into an equivalent constrained optimization, whose solution can be computed in a distributed fashion using the alternating-direction method of multipliers (Section III-A). The final LMS-type recursions are motivated from stochastic approximation techniques, while we illustrate the intuition and flexibility of the resulting algorithm (Section III-B). A first-order (convergence in the mean) analysis is presented in Section IV while mss convergence and tracking capabilities are corroborated with the aid of simulations (Section V). We finally conclude this paper in Section VI.

Notation: Bold uppercase letters will denote matrices (ij-th entry denoted by $[.]_{ij}$) whereas bold lowercase letters denote column vectors (i-th entry denoted by $[.]_i$); $\otimes$, $(.)^T$, diag$(.)$, bdiag$(.)$, $E\,[.]$ denote Kronecker product, transposition, diagonal matrix (arguments are scalar diagonal entries), block diagonal matrix (arguments are matrix diagonal entries) and expectation, respectively. For a vector, $\|.\|$ corresponds to its Euclidean norm and for a set $|.|$ denotes its cardinality. The $n \times n$ identity matrix is denoted by $\mathbf{I}_n$.

II. PRELIMINARIES AND PROBLEM STATEMENT

Consider an ad hoc WSN comprising J sensors, where only single-hop communications are allowed, i.e., sensor j can only communicate with the sensors in its neighborhood $\mathcal{N}_j \subseteq [1, J]$. The communication links are assumed to be symmetric, and the WSN is modelled as an undirected graph whose vertices are the sensors and its edges represent the available links. Global connectivity information is captured by the symmetric adjacency matrix $\mathbf{E} \in \mathbb{R}^{J\times J}$, where $[\mathbf{E}]_{ij} = 1$ if $i \in \mathcal{N}_j$ and 0 otherwise. As in [8], the communication graph is assumed to be connected; i.e., there exists a (possibly) multi-hop communication path connecting *any* two sensors. An example of such a network is shown on Fig. 1.

† Prepared through collaborative participation in the Communications and Networks Consortium sponsored by the U. S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011. The U. S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

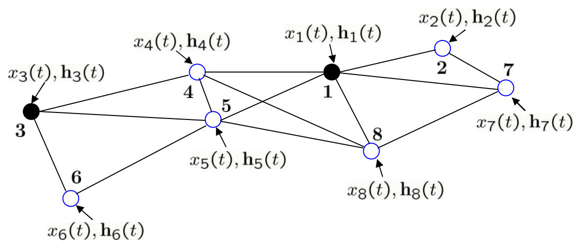


Fig. 1. An ad hoc WSN with 8 sensors.

The WSN is deployed to estimate a signal vector $\mathbf{s}_0 \in \mathbb{R}^{p\times 1}$. For every given time instant t (t denotes discrete time), each sensor acquires a regressor vector $\mathbf{h}_j(t) \in \mathbb{R}^{p\times 1}$ and a scalar observation $x_j(t)$, both assumed zero-mean without loss of generality. Furthermore, it is assumed that both regressors and observations are independent across space (sensors) whereas arbitrary time correlations are allowed. Introducing the global quantities $\mathbf{x}(t) := [x_1(t), \ldots, x_J(t)]^T \in \mathbb{R}^{J\times 1}$ and $\mathbf{H}(t) := [\mathbf{h}_1(t), \ldots, \mathbf{h}_J(t)]^T \in \mathbb{R}^{J\times p}$, the minimum mean-squared error (MMSE) estimation problem of interest can be posed as

$$\begin{aligned}\hat{\mathbf{s}}(t) &= \arg\min_{\mathbf{s}} E\left[\|\mathbf{x}(t) - \mathbf{H}(t)\mathbf{s}\|^2\right] \\ &= \arg\min_{\mathbf{s}} \sum_{j=1}^{J} E\left[(x_j(t) - \mathbf{h}_j^T(t)\mathbf{s})^2\right]. \end{aligned} \quad (1)$$

If $\mathbf{x}(t)$ and $\mathbf{H}(t)$ are jointly stationary, then (1) leads to the well-known Wiener solution $\mathbf{s}_0 = \mathbf{K}_H^{-1}\mathbf{K}_{Hx}$ [9, p.15], where $\mathbf{K}_H = E\left[\mathbf{H}^T(t)\mathbf{H}(t)\right]$ and $\mathbf{K}_{Hx} = E\left[\mathbf{H}^T(t)\mathbf{x}(t)\right]$. If global (cross-) covariance matrices $\mathbf{K}_H$ and $\mathbf{K}_{Hx}$ were available, then a steepest-descent iterative algorithm with sufficiently small step-size would converge to $\mathbf{s}_0$ while avoiding the matrix inversion overhead. In many applications of interest, this statistical information will not be available and one could resort to the centralized (C-) LMS algorithm; see e.g., [9, p.77]

$$\hat{\mathbf{s}}(t) = \hat{\mathbf{s}}(t-1) + \mu\mathbf{H}^T(t)\left[\mathbf{x}(t) - \mathbf{H}(t)\hat{\mathbf{s}}(t-1)\right]. \quad (2)$$

Considering a constant step-size μ in order to allow for tracking of a possibly time-varying $\mathbf{s}_0(t)$, the C-LMS algorithm will yield stochastic iterates $\hat{\mathbf{s}}(t)$ that will not converge to, but hover around $\mathbf{s}_0$. In fact, it can be shown [9, Ch.5, Ch.9] that for appropriately chosen step-sizes and observations obeying a linear model, i.e., $x_j(t) = \mathbf{h}_j(t)\mathbf{s}_0 + \epsilon_j(t)$ with $\mathbf{h}_j(t)$ independent of the white noise $\epsilon_j(t)$; recursion (2): (i) provides asymptotically unbiased estimates, i.e., $\lim_{t\to\infty} E\left[\hat{\mathbf{s}}(t)\right] = \mathbf{s}_0$, (ii) yields an asymptotic error covariance matrix $\lim_{t\to\infty} E\left[(\hat{\mathbf{s}}(t) - \mathbf{s}_0)(\hat{\mathbf{s}}(t) - \mathbf{s}_0)^T\right]$ with bounded entries; and, (iii) yields an asymptotically inflated MSE, i.e., $\lim_{t\to\infty} E\left[\|\mathbf{x}(t) - \mathbf{H}(t)\hat{\mathbf{s}}(t)\|^2\right] = J_{\min} + \alpha(\mu)$, where $J_{\min} = E\left[\|\mathbf{x}(t) - \mathbf{H}(t)\mathbf{s}_0\|^2\right]$ and $\alpha(\mu) > 0$ is the so called *excess mean-square error*.

Remark 1: The C-LMS algorithm establishes a performance benchmark among the LMS-type adaptation rules, as every update encompasses all the information available in the network. Although both the observations $\mathbf{x}(t)$ and regressor columns in $\mathbf{H}(t)$ are actually disseminated across the WSN, the C-LMS can be implemented in a fusion center (FC) based topology. This, however, comes at the price of isolating the network's point of failure and increasing the communication cost (thus diminishing sensor battery lifetime).

For these reasons, the purpose of this paper is to derive a fully distributed LMS algorithm using ad hoc WSNs whose performance is comparable to its centralized counterpart (2). In a nutshell, the described setup naturally suggests the following characteristics that the algorithm should exhibit: (i) local sensor estimates should eventually reach consensus, even in the presence of receiver noise, in the sense that they should all enjoy the previously mentioned asymptotic properties of the C-LMS, (ii) processing at the sensor level should be kept as simple as possible; and, (iii) information should be exchanged between neighboring sensors only.

III. The D-LMS Algorithm

In this section we introduce the D-LMS algorithm, first going through the detailed process of algorithm construction and describing its operation. We also reinterpret the resulting recursions so as to draw conclusions on how local and network-wide information is used in the learning process, and understand the mechanisms employed to reach consensus.

A. Algorithm Construction

Noting that the global variable $\mathbf{s}$ couples the summands on the cost function in the right-hand side (rhs) of (1), we introduce the set of auxiliary variables $\{\mathbf{s}_j\}_{j=1}^J$ that represent the local estimates at each of the sensors. To this end, we consider the *constrained* minimization problem

$$\begin{aligned}\{\hat{\mathbf{s}}_j(t)\}_{j=1}^J &= \arg\min_{\mathbf{s}_j} \sum_{j=1}^{J} E\left[(x_j(t) - \mathbf{h}_j^T(t)\mathbf{s}_j)^2\right] \\ &\text{s. t.} \quad \varepsilon_j\mathbf{s}_j = \varepsilon_j\bar{\mathbf{s}}_b,\; b\in\mathcal{B}, j\in\mathcal{N}_b, \end{aligned} \quad (3)$$

where $\mathcal{B} \subseteq [1, J]$ is the bridge sensor subset introduced in [8] which is defined by the following pair of conditions: (i) $\forall\, j \in [1, J]$ there exists at least one $b \in \mathcal{B}$ such that $b \in \mathcal{N}_j$ (the bridge neighbors of sensor j will be denoted by $\mathcal{B}_j := \mathcal{N}_j \cap \mathcal{B}$); and, (ii) $\forall\, b_1 \in \mathcal{B}$ there exists another sensor $b_2 \in \mathcal{B}$ such that the shortest path between b_1 and b_2 has at most two edges. Note that the set of all sensors $[1, J]$ is a valid bridge sensor subset, but simple algorithms can determine other possible choices with smaller cardinality (see also Fig. 1 where $\mathcal{B}$ comprises the sensors in black). Useful in our derivations is the non-symmetric bridge adjacency matrix $\mathbf{E}_{\mathcal{B}} \in \mathbb{R}^{J\times J}$, with $[\mathbf{E}_{\mathcal{B}}]_{ij} = 1$ if $j \in \mathcal{B}_i$ and 0 otherwise. The additional set of consensus-enforcing variables $\{\bar{\mathbf{s}}_b\}_{b\in\mathcal{B}}$ are maintained at each of the bridge sensors, whereas the WSN connectivity assumption plus the defining characteristics of $\mathcal{B}$ provide necessary and sufficient conditions to assure that the equality constraints in (3) imply $\mathbf{s}_j = \mathbf{s}_{j'}\ \forall\, j, j' \in [1, J]$ [8, Proposition 1]. This establishes the equivalence of (1) and (3) in the sense that $\hat{\mathbf{s}}_j(t) = \hat{\mathbf{s}}(t)\ \forall\, j \in [1, J]$. Regarding the positive constants ε_j, though they do not cause any effect whatsoever on the constraints in (3), they will play an important role in the convergence characteristics of the D-LMS algorithm (see Remark 3).

In order to solve (3), we consider its augmented Lagrangian given by

$$\begin{aligned}\mathcal{L}_a\left[\mathbf{s}, \bar{\mathbf{s}}, \mathbf{v}\right] = &\sum_{j=1}^{J} E\left[(x_j(t+1) - \mathbf{h}_j^T(t+1)\mathbf{s}_j)^2\right] \\ &+ \sum_{b\in\mathcal{B}}\sum_{j\in\mathcal{N}_b} \varepsilon_j \left(\mathbf{v}_j^b\right)^T(\mathbf{s}_j - \bar{\mathbf{s}}_b) \\ &+ \sum_{b\in\mathcal{B}}\sum_{j\in\mathcal{N}_b} \frac{c_j\varepsilon_j^2}{2}\|\mathbf{s}_j - \bar{\mathbf{s}}_b\|^2, \end{aligned} \quad (4)$$

where $\mathbf{s} := \{\mathbf{s}_j\}_{j=1}^J$, $\bar{\mathbf{s}} := \{\bar{\mathbf{s}}_b\}_{b\in\mathcal{B}}$, $\mathbf{v} := \left\{\mathbf{v}_j^b\right\}_{j\in[1,J]}^{b\in\mathcal{B}_j}$ comprises the Lagrange multiplier vectors, and $c_j > 0$ are penalty coefficients corresponding to the constraint $\varepsilon_j\mathbf{s}_j = \varepsilon_j\bar{\mathbf{s}}_b\ ,\forall\, b \in \mathcal{B}$. We will now resort to the alternating-direction method of Lagrange multipliers [1, p.253] to iteratively solve (3), obtaining a set of simple recursions to update $\{\mathbf{s}, \bar{\mathbf{s}}, \mathbf{v}\}$ in a purely distributed fashion. Because the algorithm is designed for online estimation applications, the recursions will run in real-time and hence iteration indexes will coincide with the time index t. First, the local recursions to update the Lagrange multipliers are given by [1]

$$\mathbf{v}_j^b(t) = \mathbf{v}_j^b(t-1) + \varepsilon_j c_j\left(\mathbf{s}_j(t) - \bar{\mathbf{s}}_b(t)\right), \quad j \in [1, J],\ b \in \mathcal{B}_j \quad (5)$$

Secondly, the recursions for the local estimates $\mathbf{s}_j$ are obtained by minimizing (4) using block coordinate descent, i.e., $\mathcal{L}_a[.]$ is minimized with regards to $\mathbf{s}_j$ assuming all other variables $\{\bar{\mathbf{s}}_b(t)\}_{b\in\mathcal{B}}$, $\left\{\mathbf{v}_j^b(t)\right\}_{j\in[1,J]}^{b\in\mathcal{B}_j}$ are fixed (see e.g. [8, Appendix B]). Because (4) is convex, the first-order necessary condition is also sufficient for optimality. Computing the gradient with respect to $\mathbf{s}_j$ and setting the result equal to zero, yields the desired solution as the root of

$$\begin{aligned}E[&-2\mathbf{h}_j(t+1)\left(x_j(t+1) - \mathbf{h}_j^T(t+1)\mathbf{s}_j\right) \\ &+ \sum_{b\in\mathcal{B}_j}\varepsilon_j\mathbf{v}_j^b(t) + \sum_{b\in\mathcal{B}_j}\varepsilon_j^2 c_j\left(\mathbf{s}_j - \bar{\mathbf{s}}_b(t)\right)] = \mathbf{0}\end{aligned}$$

where we have moved the summations of the deterministic quantities inside the expectation. In the absence of the statistical information $E\left[\mathbf{h}_j(t+1)x_j(t+1)\right]$ and $E\left[\mathbf{h}_j(t+1)\mathbf{h}_j^T(t+1)\right]$, the root cannot be computed in closed form. Thus, motivated by stochastic approximation techniques (e.g. the Robbins-Monro algorithm) [4, Ch.1] to find the roots of an unknown function given noisy observations, the proposed recursion for $j \in [1, J]$ is

$$\begin{aligned}\mathbf{s}_j(t+1) = \mathbf{s}_j(t) + \mu_j[&2\mathbf{h}_j(t+1)e_j(t+1) - \varepsilon_j^2 c_j|\mathcal{B}_j|\mathbf{s}_j(t) \\ &- \sum_{b\in\mathcal{B}_j}\left(\varepsilon_j\mathbf{v}_j^b(t) - \varepsilon_j^2 c_j\bar{\mathbf{s}}_b(t)\right)],\end{aligned} \quad (6)$$

where μ_j is a constant step-size and $e_j(t+1) := x_j(t+1) - \mathbf{h}_j(t+1)^T\mathbf{s}_j(t)$ is the local prior error. The update equations for the consensus imposing variables $\bar{\mathbf{s}}_b$ are obtained similarly to (6), minimizing (4) with fixed $\left\{\mathbf{s}_j(t+1), \mathbf{v}_j^b(t)\right\}_{j\in[1,J]}^{b\in\mathcal{B}_j}$ and noting that the expectation term is a constant with respect to $\bar{\mathbf{s}}_b$. This yields for $b \in \mathcal{B}$

$$\bar{\mathbf{s}}_b(t+1) = \sum_{j\in\mathcal{N}_b}\frac{\varepsilon_j\mathbf{v}_j^b(t) + \varepsilon_j^2 c_j\mathbf{s}_j(t+1)}{\sum_{r\in\mathcal{N}_b}\varepsilon_r^2 c_r}. \quad (7)$$

Recursions (5)-(7) constitute the D-LMS algorithm, where initial conditions can be arbitrary. At time instant t, sensor j receives the consensus variables $\bar{\mathbf{s}}_b(t)$ from its bridge neighbors $b \in \mathcal{B}_j$. With this information and using (5), it is able to update its Lagrange multipliers $\left\{\mathbf{v}_j^b(t)\right\}_{b\in\mathcal{B}_j}$ which are then used to compute $\mathbf{s}_j(t+1)$ via (6). Then sensor j transmits the vector $\varepsilon_j\mathbf{v}_j^b(t) + \varepsilon_j^2 c_j\mathbf{s}_j(t+1)$ to all bridge sensors in its neighborhood $\mathcal{B}_j$. Consequently, each sensor $b \in \mathcal{B}$ receives the vectors $\left\{\varepsilon_j\mathbf{v}_j^b(t) + \varepsilon_j^2 c_j\mathbf{s}_j(t+1)\right\}_{j\in\mathcal{N}_b}$ whose weighted average is computed using (7) to yield $\bar{\mathbf{s}}_b(t+1)$ completing the $t-$th iteration. Communication cost is $\mathcal{O}(p)$ per iteration. Further, observe that in order to compute the weights in (7), bridge sensor b should acquire $\left\{\varepsilon_j^2 c_j\right\}_{j\in\mathcal{N}_b}$ only from its neighbors during the start-up phase of the WSN.

Having described the required single-hop exchanges of information among sensors, it can be readily appreciated how recursions (5)-(7) need to be modified in order to account for links corrupted in the presence of additive communication noise (e.g., quantization, or, reception noise). For all sensors $j \in [1, J]$, $b \in \mathcal{B}_j$ in (8) and $b \in \mathcal{B}$ in (10), the D-LMS algorithm in the noisy setup becomes

$$\mathbf{v}_j^b(t) = \mathbf{v}_j^b(t-1) + \varepsilon_j c_j\left(\mathbf{s}_j(t) - (\bar{\mathbf{s}}_b(t) + \mathbf{n}_j^b(t))\right), \quad (8)$$

$$\begin{aligned}\mathbf{s}_j(t+1) = \mathbf{s}_j(t) + \mu_j[&2\mathbf{h}_j(t+1)e_j(t+1) - \varepsilon_j^2 c_j|\mathcal{B}_j|\mathbf{s}_j(t) \\ &- \sum_{b\in\mathcal{B}_j}\left(\varepsilon_j\mathbf{v}_j^b(t) - \varepsilon_j^2 c_j(\bar{\mathbf{s}}_b(t) + \mathbf{n}_j^b(t))\right)],\end{aligned} \quad (9)$$

$$\bar{\mathbf{s}}_b(t+1) = \sum_{j\in\mathcal{N}_b}\frac{\varepsilon_j\mathbf{v}_j^b(t) + \varepsilon_j^2 c_j\mathbf{s}_j(t+1) + \mathbf{n}_b^j(t+1)}{\sum_{r\in\mathcal{N}_b}\varepsilon_r^2 c_r} \quad (10)$$

where $\mathbf{n}_j^i(t)$ denotes a zero-mean additive noise vector corrupting a variable transmitted from sensor i to sensor j, at time instant t.

Remark 2: The bridge sensor set provides the flexibility to trade-off communication cost for robustness to sensor failures. In the sample network of Fig. 1, on a per iteration basis D-LMS requires approximately half the number of inter-sensor variable exchanges than diffusion LMS [5]. If on the other hand $\mathcal{B} \equiv [1, J]$, the situation is reversed. Regarding recovery from sensor failures, D-LMS requires each sensor to adjust its local recursions (8)-(10) to the modified neighborhood structure.

B. Further Insights

Even though recursions (5)-(7) clearly suggest simplicity as an asset of the proposed algorithm, they may somehow obscure the essential mechanisms operating on the available information to yield the estimates $\mathbf{s}_j$. Here we derive a set of equivalent recursions that turn out to be insightful about these issues.

For arbitrary $j \in [1, J]$ and $b \in \mathcal{B}_j$, consider the Lagrange multiplier update recursion (5) with initial condition $\mathbf{v}_j^b(-1) = \mathbf{0}$. Repeated substitutions of the resulting iterates into the rhs of (5) leads to

$$\mathbf{v}_j^b(t) = \sum_{n=0}^{t} \varepsilon_j c_j \left[\mathbf{s}_j(n) - \bar{\mathbf{s}}_b(n)\right]. \tag{11}$$

Using (11) to eliminate $\mathbf{v}_j^b(t)$ from (7) we obtain for all $b \in \mathcal{B}$ and $t \geq 0$

$$\bar{\mathbf{s}}_b(t) = \sum_{j \in \mathcal{N}_b} \frac{\varepsilon_j^2 c_j \mathbf{s}_j(t)}{\sum_{r \in \mathcal{N}_b} \varepsilon_r^2 c_r}. \tag{12}$$

Equation (12) establishes that the consensus variables $\bar{\mathbf{s}}_b(t)$ are simply obtained as a weighted average of the local estimates gathered from sensor b's neighborhood.

To this end, consider the vector quantity $\mathbf{q}_j(t) := \mathbf{s}_j(t) - |\mathcal{B}_j|^{-1} \sum_{b \in \mathcal{B}_j} \bar{\mathbf{s}}_b(t)$ which represents the *instantaneous consensus error* at sensor j, as measured with respect to the consensus reference given by the average $|\mathcal{B}_j|^{-1} \sum_{b \in \mathcal{B}_j} \bar{\mathbf{s}}_b(t)$. Setting the penalty coefficients as $c_j = \gamma_j / |\mathcal{B}_j|$ (with $\gamma_j = 1$ in what follows to simplify notation), using (11) to eliminate the Lagrange multipliers from (6) and rearranging terms yields

$$\begin{aligned}\mathbf{s}_j(t+1) = &\overbrace{\mathbf{s}_j(t) + \mu_j 2\mathbf{h}_j(t+1)e(t+1)}^{(a)} \\ &- \overbrace{\mu_j \varepsilon_j^2 \mathbf{q}_j(t)}^{(b)} - \overbrace{\mu_j \varepsilon_j^2 \sum_{n=0}^{t} \mathbf{q}_j(n)}^{(c)}.\end{aligned} \tag{13}$$

The pair (12)-(13) is equivalent to the D-LMS under ideal links, when zero initial conditions are chosen for the Lagrange multipliers. Also note that the new recursions are not suitable for implementation in real-time as the computation of (c) in (13) requires storing the whole history of $\mathbf{q}_j(t)$. Nonetheless, they shed further light onto the signal processing taking place at each sensor, which turns out to be remarkably intuitive as we discuss next.

As expressed in (13), the recursion suggests that the local estimate $\mathbf{s}_j(t+1)$ is obtained as the superposition of three terms: (a) a purely *local* LMS update based on the new information $\{\mathbf{h}_j(t+1), x_j(t+1)\}$ available at sensor j; (b) an update based on a proportional correction due to the instantaneous consensus error $\mathbf{q}_j(t)$; and, (c) a correction due to the accumulated consensus error (discrete-time integral). A term like (a) is pleasing as well as expected, whereas the rest should explain the mechanisms employed to incorporate the extra information gathered throughout the whole WSN. In fact, (b) and (c) show that a proportional-integral (PI) discrete-time controller [2, p.605] is used to drive the local estimate $\mathbf{s}_j(t)$ to consensus, as dictated by the computed set-point $|\mathcal{B}_j|^{-1} \sum_{b \in \mathcal{B}_j} \bar{\mathbf{s}}_b(t)$ (see Fig. 2). It is exclusively throughout this reference, that all the information obtained from the network is introduced in the update of $\mathbf{s}_j$.

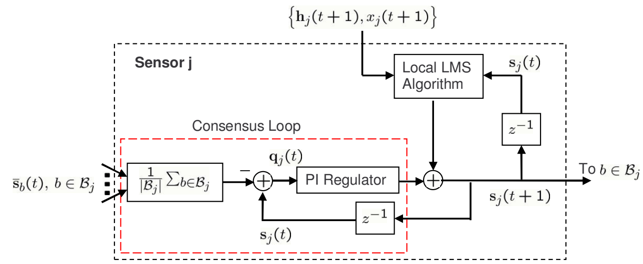


Fig. 2. D-LMS processing at the sensor level.

Examination of (12) shows that for every time slot, the information range of the D-LMS has a radius of two hops; as neighbors of $b \in \mathcal{B}_j$ may be up to two hops away from sensor j. This should be contrasted with the diffusion LMS [5], whose instantaneous information range spans only the single-hop neighborhood.

Remark 3: The constant ε_j is only affecting the proportional and integral gains of the consensus regulator [cf. (13)]. For $\varepsilon_j = 1$ these gains boil down to μ_j, a generally small constant that will attenuate the influence that the information embedded in (b) and (c) has on the estimate $\mathbf{s}_j(t+1)$. The presence of ε_j is thus intuitively justified so as to compensate for this effect, gaining an additional degree of freedom to attain potentially higher convergence rates.

IV. Mean Analysis

Here we focus on the D-LMS algorithm analysis, by establishing its convergence in the mean under the *independence* setting. This setting is characterized by the following signal assumptions for all $j \in [1, J]$:

- **(As1)** $\mathbf{h}_j(t)$ is a zero-mean white random vector with positive definite covariance matrix $\mathbf{K}_{h_j} = E[\mathbf{h}_j(t)\mathbf{h}_j(t)^T]$, whose spectral radius will be denoted by $\lambda_j^{\max}$.
- **(As2)** Observations obey the linear model $x_j(t) = \mathbf{h}_j(t)^T \mathbf{s}_0 + \epsilon_j(t)$, where $\epsilon_j(t)$ is a zero-mean white noise.
- **(As3)** $\mathbf{h}_j(t)$ and $\epsilon_j(t)$ are statistically independent.

Defining $\mathbf{m}_j(t) := E\left[\mathbf{s}_j(t) - \mathbf{s}_0\right]$, we will derive sufficient conditions under which the global averaged error vector $\mathbf{m}(t) := \left[\mathbf{m}_1(t)^T, \ldots, \mathbf{m}_J(t)^T\right]^T \in \mathbb{R}^{Jp \times 1}$ satisfies $\lim_{t \to \infty} \mathbf{m}(t) = \mathbf{0}$. As established in the following Lemma, $\mathbf{m}(t)$ is the solution of a second-order homogenous vector difference equation with specific initial conditions.

Lemma 1: *Under assumptions (As1)-(As3), consider the D-LMS algorithm (8)-(10) initialized with* $\{\mathbf{s}_j(0)=\mathbf{0}\}_{j\in[1,J]}$, $\{\mathbf{v}_j^b(-1)=\mathbf{0}\}_{j\in[1,J]}^{b\in\mathcal{B}_j}$, $\{\bar{\mathbf{s}}_b(0)=\mathbf{0}\}_{b\in\mathcal{B}}$. *Then for* $t\geq 1$, $\mathbf{m}(t+1)$ *is given by the second order recursion* $\mathbf{m}(t+1)=\mathbf{A}\mathbf{m}(t)+\mathbf{B}\mathbf{m}(t-1)$ *with* $\mathbf{m}(0)=-\left[\mathbf{s}_0^T,\ldots,\mathbf{s}_0^T\right]^T$ *and* $\mathbf{m}(1)=\left[\mathbf{s}_0^T\left(\mu_1 2\mathbf{K}_{h_1}-\mathbf{I}_p\right),\ldots,\mathbf{s}_0^T\left(\mu_J 2\mathbf{K}_{h_J}-\mathbf{I}_p\right)\right]^T$, *where*

$$\mathbf{A}:=\text{bdiag}\big(2(1-\mu_1\varepsilon_1^2c_1|\mathcal{B}_1|)\mathbf{I}_p-\mu_1 2\mathbf{K}_{h_1},\ldots,\\ 2(1-\mu_J\varepsilon_J^2c_J|\mathcal{B}_J|)\mathbf{I}_p-\mu_J 2\mathbf{K}_{h_J}\big)+2\mathbf{W} \quad (14)$$

$$\mathbf{B}:=-\text{bdiag}\big((1-\mu_1\varepsilon_1^2c_1|\mathcal{B}_1|)\mathbf{I}_p-\mu_1 2\mathbf{K}_{h_1},\ldots,\\ (1-\mu_J\varepsilon_J^2c_J|\mathcal{B}_J|)\mathbf{I}_p-\mu_J 2\mathbf{K}_{h_J}\big)-\mathbf{W} \quad (15)$$

and

$$\mathbf{W}:=[\text{diag}(\mu_1\varepsilon_1^2c_1,\ldots,\mu_J\varepsilon_J^2c_J)\cdot\mathbf{E}_{\mathcal{B}}\cdot\\ \text{diag}((\sum_{r\in\mathcal{N}_1}\varepsilon_r^2c_r)^{-1},\ldots,(\sum_{r\in\mathcal{N}_J}\varepsilon_r^2c_r)^{-1})\cdot\\ \mathbf{E}\cdot\text{diag}(\varepsilon_1^2c_1,\ldots,\varepsilon_J^2c_J)]\otimes\mathbf{I}_p$$

Proof: Turning the D-LMS recursions (8)-(10) into an error form by subtracting $\mathbf{s}_0$ from $\mathbf{s}_j(t)$, and noting that by virtue of **(As2)** we can write $e_j(t+1)=\epsilon_j(t+1)-\mathbf{h}_j^T(t+1)(\mathbf{s}_j(t)-\mathbf{s}_0)$, we proceed to take expectations using all **(As1)-(As3)** and the zero-mean property of the noise vectors to obtain

$$\underline{\mathbf{v}}_j^b(t)=\underline{\mathbf{v}}_j^b(t-1)+\varepsilon_jc_j\left(\mathbf{m}_j(t)-\bar{\underline{\mathbf{s}}}_b(t)\right),\quad b\in\mathcal{B}_j \quad (16)$$

$$\mathbf{m}_j(t+1)=\mathbf{m}_j(t)+\mu_j[-2\mathbf{K}_{h_j}\mathbf{m}_j(t)-\varepsilon_j^2c_j|\mathcal{B}_j|\mathbf{m}_j(t)\\ -\sum_{b\in\mathcal{B}_j}\left(\varepsilon_j\underline{\mathbf{v}}_j^b(t)-\varepsilon_j^2c_j\bar{\underline{\mathbf{s}}}_b(t)\right)] \quad (17)$$

$$\bar{\underline{\mathbf{s}}}_b(t+1)=\sum_{j\in\mathcal{N}_b}\frac{\varepsilon_j\underline{\mathbf{v}}_j^b(t)+\varepsilon_j^2c_j\mathbf{m}_j(t+1)}{\sum_{r\in\mathcal{N}_b}\varepsilon_r^2c_r}, \quad (18)$$

where $\{\underline{\mathbf{v}}_j^b\}_{j\in[1,J]}^{b\in\mathcal{B}_j}$, $\{\bar{\underline{\mathbf{s}}}_b\}_{b\in\mathcal{B}}$ stand for the averaged Lagrange multipliers and consensus variables respectively, and the initial conditions are $\{\mathbf{m}_j(0)=-\mathbf{s}_0\}_{j\in[1,J]}$, $\{\underline{\mathbf{v}}_j^b(-1)=\mathbf{0}\}_{j\in[1,J]}^{b\in\mathcal{B}_j}$, $\{\bar{\underline{\mathbf{s}}}_b(0)=\mathbf{0}\}_{b\in\mathcal{B}}$. Recursions (16),(18) have the same form as (5),(7) and therefore expressions (11)-(12) are still valid for the averaged variables. Using them both to eliminate the averaged multipliers and consensus variables from (17), and arguing by induction after subtracting the resulting recursion for $\mathbf{m}_j(t)$ from the one for $\mathbf{m}_j(t+1)$ allows to conclude that for all $j\in[1,J]$, the second order recursion

$$\mathbf{m}_j(t+1)=[2(1-\mu_j\varepsilon_j^2c_j|\mathcal{B}_j|)\mathbf{I}_p-\mu_j2\mathbf{K}_{h_j}]\mathbf{m}_j(t)\\ -[(1-\mu_j\varepsilon_j^2c_j|\mathcal{B}_j|)\mathbf{I}_p-\mu_j2\mathbf{K}_{h_j}]\mathbf{m}_j(t-1)\\ +\sum_{b\in\mathcal{B}_j}\frac{\mu_j\varepsilon_j^2c_j}{\sum_{r\in\mathcal{N}_b}\varepsilon_r^2c_r}\sum_{i\in\mathcal{N}_b}\varepsilon_i^2c_i2\mathbf{m}_i(t)\\ -\sum_{b\in\mathcal{B}_j}\frac{\mu_j\varepsilon_j^2c_j}{\sum_{r\in\mathcal{N}_b}\varepsilon_r^2c_r}\sum_{i\in\mathcal{N}_b}\varepsilon_i^2c_i\mathbf{m}_i(t-1), \quad (19)$$

initialized with $\{\mathbf{m}_j(0)=-\mathbf{s}_0\}_{j\in[1,J]}$, $\{\mathbf{m}_j(1)=(\mu_j2\mathbf{K}_{h_j}-\mathbf{I}_p)\mathbf{s}_0\}_{j\in[1,J]}$, is equivalent for $t\geq 1$ to the averaged system (16)-(18). Concatenating the local variables to form $\mathbf{m}(t)$, the set of recursions (19) for $j=1,\ldots,J$ can be readily written in compact form as $\mathbf{m}(t+1)=\mathbf{A}\mathbf{m}(t)+\mathbf{B}\mathbf{m}(t-1)$ with $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{W}$ as defined in the statement of the Lemma, and with the required initial conditions. This completes the proof of Lemma 1. □

Observe how the structure of matrices $\mathbf{A}$ and $\mathbf{B}$ [cf. (14)-(15)] shows that the local averaged error $\mathbf{m}_j(t+1)$ not only depends upon its past values $\mathbf{m}_j(t)$, $\mathbf{m}_j(t-1)$ through the block diagonal terms, but also upon the past values $\mathbf{m}_i(t)$, $\mathbf{m}_i(t-1)$ of the sensors $i\in\mathcal{N}_b$, with $b\in\mathcal{B}_j$. This last network-wide coupling comes through the matrix $\mathbf{W}$, which introduces network interactions through the adjacency matrices $\mathbf{E}$ and $\mathbf{E}_{\mathcal{B}}$.

In order to study the convergence of the second-order recursion, we consider the equivalent first-order system obtained by state concatenation

$$\tilde{\mathbf{m}}(t+1):=\begin{bmatrix}\mathbf{m}(t+1)\\ \mathbf{m}(t)\end{bmatrix}=\mathbf{\Phi}\tilde{\mathbf{m}}(t), \quad (20)$$
$$\mathbf{\Phi}:=\begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{I}_{Jp} & \mathbf{0}\end{bmatrix}\in\mathbb{R}^{2Jp\times 2Jp}$$

and initial condition $\tilde{\mathbf{m}}(1)=[\mathbf{m}(1)^T,\mathbf{m}(0)^T]^T$ as defined in Lemma 1. The convergence of (20) will be intimately related to the eigenstructure of $\mathbf{\Phi}$, leading us to the first-order characterization of the D-LMS algorithm that follows.

Proposition 1: *Under assumptions (As1)-(As3), the D-LMS algorithm (8)-(10) whose positive step-sizes* $\{\mu_j\}_{j\in[1,J]}$ *and relevant parameters are chosen such that* $\mu_j(\varepsilon_j^2c_j|\mathcal{B}_j|+2\lambda_j^{\max})<1$, *achieves consensus in the mean sense i.e.,*

$$\lim_{t\to\infty}E\left[\mathbf{s}_j(t)-\mathbf{s}_0\right]=\lim_{t\to\infty}\mathbf{m}_j(t)=\mathbf{0},\quad\forall\, j\in[1,J]. \quad (21)$$

Proof: The desired result is a consequence of $\mathbf{\Phi}$'s eigenstructure properties stated in the following Lemma, which can be proved by mimicking the steps in [8, Lemma 6].

Lemma 2: *If* $\mu_j(\varepsilon_j^2c_j|\mathcal{B}_j|+2\lambda_j^{\max})<1$ *for* $j=1,\ldots,J$, *the eigenvalues* $\{\lambda_{\Phi,i}\}_{i=1}^{2Jp}$ *of* $\mathbf{\Phi}$ *and the corresponding right and left eigenvectors* $\mathbf{u}_{\Phi,i}$ *and* $\mathbf{v}_{\Phi,i}$ *satisfy the following properties:*

(a) *Exactly p eigenvalues are equal to* 1, *while the rest satisfy* $|\lambda_{\Phi,i}|<1$.

(b) $\mathbf{v}_{\Phi,i}=\left[(\mathbf{v}_{\Phi,i}^1)^T,(\mathbf{v}_{\Phi,i}^2)^T\right]^T$ *is a left eigenvector associated with the eigenvalue* 1 *if and only if the following equations hold:*

$$(\mathbf{v}_{\Phi,i}^1)^T\left(\mathbf{I}_{Jp}-\mathbf{A}\right)=(\mathbf{v}_{\Phi,i}^2)^T \quad (22)$$
$$(\mathbf{v}_{\Phi,i}^1)^T\mathbf{B}=(\mathbf{v}_{\Phi,i}^2)^T. \quad (23)$$

From (20) it follows that $\tilde{\mathbf{m}}(t+1) = \mathbf{\Phi}^t \tilde{\mathbf{m}}(1)$, and relying on the eigendecomposition of $\mathbf{\Phi}$ we can write $\mathbf{\Phi}^t = \sum_{i=1}^{2Jp} \lambda_{\Phi,i}^t \mathbf{u}_{\Phi,i} \mathbf{v}_{\Phi,i}^T$. Applying the results from Lemma 2 (a) we obtain

$$\lim_{t\to\infty} \tilde{\mathbf{m}}(t+1) = \left(\sum_{i|\lambda_{\Phi,i}=1} \mathbf{u}_{\Phi,i} \mathbf{v}_{\Phi,i}^T \right) \tilde{\mathbf{m}}(1)$$
$$= \sum_{i|\lambda_{\Phi,i}=1} \mathbf{u}_{\Phi,i} \left((\mathbf{v}_{\Phi,i}^1)^T \mathbf{m}(1) + (\mathbf{v}_{\Phi,i}^2)^T \mathbf{m}(0) \right). \tag{24}$$

The definitions of $\mathbf{m}(0)$, $\mathbf{m}(1)$, $\mathbf{A}$ and $\mathbf{B}$ in Lemma 1 in conjunction with the conditions (22)-(23) immediately lead to the conclusion that the rhs of (24) is equal to $\mathbf{0}$. □

It is important to appreciate that Proposition 1 provides conditions for the selection of the D-LMS parameters $\{\mu_j, \varepsilon_j, c_j\}_{j\in[1,J]}$; guaranteeing stability in the mean and solely requiring the knowledge of *local* information, i.e., $\lambda_j^{\max}$ and the bridge neighborhood size $|\mathcal{B}_j|$. On the other hand, the first-order analysis pursued in [5] for the diffusion LMS under the same signal assumptions yields a condition for stability which involves global network information. That condition (which is in the spirit of spectral radius of $\mathbf{\Phi}$ smaller than 1), implicitly defines the required step-sizes while a comparison with $\mu_j(\varepsilon_j^2 c_j |\mathcal{B}_j| + 2\lambda_j^{\max}) < 1$ reinforces the simplicity of our proposed approach, now from a design stage perspective.

V. Simulations

At this point, we will test the novel D-LMS so as to illustrate its convergence characteristics and establish comparisons with the diffusion LMS with Metropolis weights in [5], [7], the C-LMS and the purely local (L-) LMS with no communication among sensors. We consider an ad hoc WSN with $J = 30$ sensors. The regressor vectors $\mathbf{h}_j(t)$ are chosen with i.i.d. $\mathcal{N}(0,1)$ entries ($\lambda_j^{\max} = 1$), whereas for the observations a linear Gaussian model $\mathbf{x}(t) = \mathbf{H}(t)\mathbf{s}_0 + \epsilon(t)$ [3] is adopted with $\sigma_{\epsilon_j}^2 = 10^{-4}$. The signal vector dimensionality is $p = 4$.

We first compare the MSE performance of the algorithms, which as a byproduct validates Proposition 1 (convergence in the mean is necessary for mean-square convergence [9]). The same step-size $\mu = 0.01$ was chosen in all cases, and in particular for the D-LMS $c_j = 1/|\mathcal{B}_j|$ and $\varepsilon_j = 1/\sqrt{2\mu}$ for $j = 1, \ldots, J$. Receiver noise of variance $\sigma_n^2 = 10^{-4}$ was considered to test the D-LMS under non-ideal links. The global MSE evolution (learning curve) computed as $J^{-1}\sum_{j=1}^J E\left[\|\mathbf{x}(t) - \mathbf{H}(t)\mathbf{s}_j(t)\|^2\right]$ for the distributed setups is shown on Fig. 3 (a), where the averaging is taken over 50 realizations of the experiment. As expected, the distributed approaches lie in between C-LMS and L-LMS and in all noise-free cases the resulting misadjustment is negligible. Regarding D-LMS: (i) it shows a higher convergence rate than the difusion LMS; and, (ii) is also convergent in the presence of receiver noise.

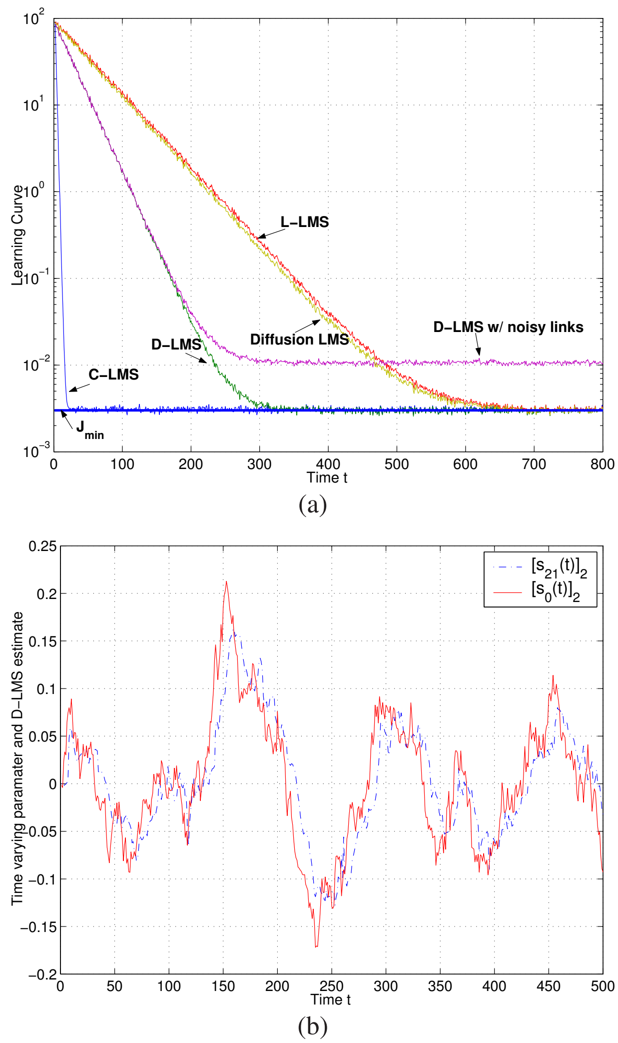
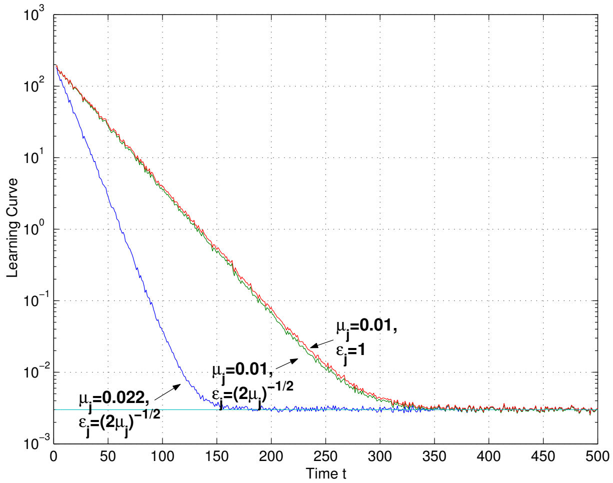


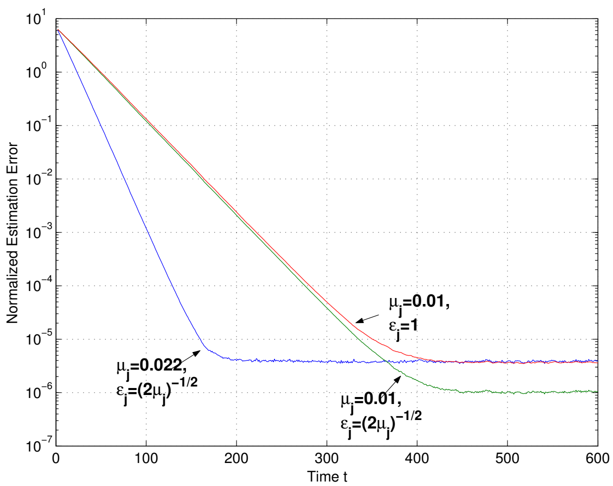
Fig. 3. *(a)* Learning curve comparisons; *(b)* Tracking with D-LMS.

We also elaborate on the flexibility and increased performance gained by the appropriate selection of ε_j, as mentioned under Remark 3. First we compare the previous setup with the case where $\varepsilon_j = 1$ and the attenuating effects of μ_j in the consensus loop gains are not compensated [cf. (13)]. Even though we observe a slight improvement in the learning curve shown in Fig. 4 (a), the major gain is manifested as a reduction of the normalized estimation error $J^{-1}\sum_{j=1}^J E\left[\|\mathbf{s}_j(t) - \mathbf{s}_0\|^2\right]$ in Fig. 4 (b). Alternatively, we adjust the step-size to the value $\mu = 0.022$ so that there is no appreciable difference in the estimation error, and observe that the compensated D-LMS shows a much faster MSE transient response (see Fig. 4).

Finally, we illustrate the tracking capabilities of the D-LMS, with $\mathbf{s}_0(t)$ given by the large-amplitude slow-speed model $\mathbf{s}_0(t) = (1-\rho)\mathbf{s}_0(t-1) + \sqrt{\rho}\nu(t)$ [9, p.127], where $\rho = 0.01$ and $\nu(t)$ is zero-mean white Gaussian with variance 0.02. Fig. 3 (b) depicts the second entry of the true time-varying parameter $[\mathbf{s}_0(t)]_2$, and the corresponding estimate from sensor 21 that closely follows the true variations.



(a)



(b)

Fig. 4. Algorithm tuning with ε_j: *(a)* Learning curve; *(b)* Normalized estimation error.

VI. Concluding Remarks

We developed a distributed LMS algorithm for operation in ad hoc WSNs, where inter-sensor links can be affected by additive noise. The simple sensor processor structure includes a local-LMS adaptation superimposed to the output of a PI regulator, which drives the local estimate to consensus as dictated by a network-wide information enriched reference. The consensus loop, if appropriately tuned, was shown to yield substantial performance gains. A condition for sensor step-size selection, based on local information and guaranteed convergence in the mean, is the main result of the first-order analysis conducted under the independence setting. Numerical examples corroborate that the D-LMS outperforms comparable online estimation schemes proposed in the literature, and is capable of tracking time-varying processes.

Stability and performance analysis of the D-LMS algorithm will be pursued for general settings using the stochastic averaging techniques in [9, Part III]. Furthermore, our algorithm construction approach can also be applied to the exponentially weighted least-squares cost, leading to a distributed recursive least-squares (D-RLS) algorithm.[1]

References

[1] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. 2nd Ed, Athena Scientific, 1999.

[2] G. F. Franklin, J. D. Powell and A. Emami-Naeini, *Feedback Control of Dynamic Systems*. 2nd Ed, Addison-Wesley, 1991.

[3] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall, 1993.

[4] H. J. Kushner and G. G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*. 2nd Ed, Springer-Verlag, 2003.

[5] C. G. Lopes and A. H. Sayed, "Diffusion least-mean-squares over adaptive networks," *Proc. ICASSP*, Honolulu, Hawaii, vol. 3, pp. 917-920, April 2007.

[6] C. H. Papadimitriou, *Computational Complexity*. Addison-Wesley, 1993.

[7] A. H. Sayed and C. G. Lopes, "Adaptive estimation algorithms over distributed networks," *Proc. 21st IEICE Signal Processing Symposium*, Kyoto, November 2006.

[8] I. D. Schizas A. Ribeiro and G. B. Giannakis, "Consensus in Ad Hoc WSNs with Noisy Links - Part I: Distributed Estimation of Deterministic Signals," *IEEE Trans. on Signal Processing*, 2007 (to appear); available at http://spincom.ece.umn.edu/.

[9] V. Solo and X. Kong, *Adaptive Signal Processing Algorithms*. Prentice Hall, 1995.

[1]The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies of the Army Research Laboratory or the U. S. Government.