[10] A. G. Akritas, "An implementation of Vincent's theorem," in *Numer. Math.*. New York: Springer-Verlag, 1980, vol. 36, pp. 53–62.

[11] V. Sharma, "Complexity of real root isolation using continued fraction," in *Proc. Int. Conf. Symbolic Algebraic Computation (ISSAC)*, 2007, pp. 339–346.

[12] J.-S. Cheng, X.-S. Gao, and C.-K. Tap, "Completenumerical isolation of real zeros in zero-dimensional triangular systems," in *Proc. ISSAC*, 2007, pp. 92–99.

[13] D. D. Siljak, "Algebraic criteria for positive realness relative to the unit circle," *J. Franklin Inst.*, vol. 295, no. 6.

Distributed Recursive Least-Squares for Consensus-Based In-Network Adaptive Estimation

Gonzalo Mateos, Ioannis D. Schizas, and Georgios B. Giannakis

Abstract—Recursive least-squares (RLS) schemes are of paramount importance for reducing complexity and memory requirements in estimating stationary signals as well as for tracking nonstationary processes, especially when the state and/or data model are not available and fast convergence rates are at a premium. To this end, a fully distributed (D-) RLS algorithm is developed for use by wireless sensor networks (WSNs) whereby sensors exchange messages with one-hop neighbors to consent on the network-wide estimates adaptively. The WSNs considered here do not necessarily possess a Hamiltonian cycle, while the inter-sensor links are challenged by communication noise. The novel algorithm is obtained after judiciously reformulating the exponentially-weighted least-squares cost into a separable form, which is then optimized via the alternating-direction method of multipliers. If powerful error control codes are utilized and communication noise is not an issue, D-RLS is modified to reduce communication overhead when compared to existing noise-unaware alternatives. Numerical simulations demonstrate that D-RLS can outperform existing approaches in terms of estimation performance and noise resilience, while it has the potential of performing efficient tracking.

Index Terms—Distributed estimation, RLS algorithm, wireless sensor networks (WSNs).

I. INTRODUCTION

It has been recognized that sensors comprising WSNs deployed to perform collaborative estimation tasks, should be empowered with signal processing tools that enable low-cost estimation of stationary signals as well as reduced-complexity tracking of nonstationary processes. Emergent WSN-based applications include distributed localization, power spectrum estimation, target tracking, and have motivated the development of *distributed adaptive* estimation schemes.

The authors are with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455 USA (e-mail: mate0058@umn.edu; schiz001@umn.edu; georgios@umn.edu).

The incremental (I-) RLS algorithm in [6] is one such approach, which sequentially incorporates new sensor data while performing least-squares (LS) estimation. In the stationary setup I-RLS yields the centralized LS estimator, at the price of requiring a Hamiltonian cycle through which local sensor estimates and matrices are continuously refined and communicated. An approximate I-RLS scheme devoid of inter-sensor matrix communications was also put forth in [6], but the NP-hard challenge of determining a Hamiltonian cycle in large-size WSNs remains [4]. Without topological constraints and increasing the degree of collaboration among sensors, a diffusion RLS algorithm was proposed in [2]. In addition to local estimates, sensors continuously diffuse raw sensor observations and regression vectors per neighborhood. This facilitates percolating new data across the WSN, but can degrade the estimation performance in the presence of communication noise. Both [2] and [6] include steady-state mean-square error (MSE) performance analysis under the independence setting assumptions [5, p. 448]. Distributed least mean-squares (LMS) counterparts are also available, trading off computational complexity for estimation performance; see [3], [8], and references therein. The space-time diffusion algorithm in [11] also allows for online incorporation of sensor information. However, it requires full knowledge of the data model and costly exchanges of matrices among neighbors in the process of consenting to the LS estimator.

The present paper develops a fully distributed (D-) RLS type of algorithm, which performs consensus-based, in-network, adaptive LS estimation. It is applicable to general ad hoc WSNs that are challenged by additive communication noise. Different from [11], D-RLS can be applied to a wide class of distributed estimation tasks as it requires no knowledge of the sensor data model. The algorithm is derived by optimizing the convex exponentially-weighted LS (EWLS) cost using distributed optimization techniques, namely the alternating-direction method of multipliers (AD-MoM) [1, p. 253]. Relative to the D-RLS variant in [7], the reformulation of the EWLS cost into separable form circumvents the requirement of the special type of sensors comprising the so-called bridge sensor subset; see also [9]. As a byproduct, this approach results in a fully distributed algorithm whereby all sensors perform the same tasks, without introducing hierarchies that may require intricate recovery protocols to cope with sensor failures. The exponential weighting effected through a forgetting factor endows D-RLS with tracking capabilities. This is desirable in a constantly changing environment, within which WSNs are envisioned to operate. Remarkably, whenever the use of powerful channel codes renders inter-sensor links virtually noise-free, the D-RLS algorithm can be streamlined to lower communication overhead, yet higher convergence rates with respect to (w.r.t.) existing approaches in [2] and [8].

Section II describes the WSN model and formulates the desired centralized exponentially-weighted least-squares estimator (EWLSE) as a convex optimization problem. A distributed power spectrum estimation task is introduced in Section II-B, to motivate the aforementioned formulation. The AD-MoM is utilized to minimize a separable reformulation of the original problem, leading to a set of local (per-sensor) recursions which constitute the D-RLS algorithm (Section III-A). After describing the operation of D-RLS, the detailed communication/computational cost analysis in Section III-B contrasts D-RLS with the existing approaches in [2], [11], and [6]. Numerical tests showcase the merits of the novel distributed estimation algorithm (Section IV), while concluding remarks are given in Section V.

II. PROBLEM STATEMENT

Consider a WSN with sensors $\{1, \ldots, J\} := \mathcal{J}$. Only single-hop communications are allowed, i.e., sensor j can communicate only with

the sensors in its neighborhood $\mathcal{N}_j \subseteq \mathcal{J}$, having cardinality $|\mathcal{N}_j|$. Assuming that inter-sensor links are symmetric, the WSN is modeled as an undirected connected graph. Different from [2], [11] and [6], the present network model accounts explicitly for non-ideal sensor-to-sensor links. Specifically, signals received at sensor j from sensor i at discrete-time instant t are corrupted by a zero-mean additive noise vector $\boldsymbol{\eta}_j^i(t)$, assumed temporally and spatially uncorrelated.

The WSN is deployed to estimate a real signal vector $\mathbf{s}_0(t) \in \mathbb{R}^{p \times 1}$ in a distributed fashion and subject to the single-hop communication constraints, by resorting to the LS criterion [5, p. 658]. Per time instant $t = 0, 1, \ldots$, each sensor acquires a regression 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. A similar data setting was considered in [2] and [6]. Given new data sequentially acquired, a pertinent approach is to consider the EWLSE [2], [5], [6]

$$\hat{\mathbf{s}}_{\mathrm{ewls}}(t) := \arg \min_{\mathbf{s}} \sum_{\tau=0}^t \sum_{j=1}^J \lambda^{t-\tau} \left[x_j(\tau) - \mathbf{h}_j^T(\tau)\mathbf{s} \right]^2 + \lambda^t \mathbf{s}^T \boldsymbol{\Phi}_0 \mathbf{s} \quad (1)$$

where $\lambda \in (0, 1]$ is a forgetting factor, while the positive definite matrix $\boldsymbol{\Phi}_0$ is included for regularization. Note that in forming the EWLSE at time t, the entire history of data $\{x_j(\tau), \mathbf{h}_j(\tau)\}_{\tau=0}^t$, $\forall j \in \mathcal{J}$ is incorporated in the online estimation process. Whenever $\lambda < 1$, past data are exponentially discarded thus enabling tracking of nonstationary processes.

Remark 1: If one can afford constructing/maintaining a cyclic path across sensors; or, having sensors continuously communicate their new data to a central unit (fusion center), then the I-RLS algorithm in [6] can find the centralized EWLSE benchmark. However, in-network (or diffusion) estimators may consume less power relative to I-RLS while exhibiting improved resilience to sensor failures—a feature particularly critical as the WSN size increases.

Next, we describe an application setup for distributed adaptive linear LS estimation, which naturally gives rise to the aforementioned data setting and highlights the importance of the problem addressed.

A. Distributed Power Spectrum Estimation

Consider an ad hoc WSN deployed e.g., for collaborative habitat monitoring, whereby sensors observe a narrowband source to determine its spectral peaks. Such information enables the WSN to disclose hidden periodicities due to a physical phenomenon controlled by e.g., a seismic source. Let $\theta(t)$ denote the source of interest, which can be modeled as an autoregressive (AR) process [10, p. 106]

$$\theta(t) = -\sum_{\tau=1}^p \alpha_\tau \theta(t - \tau) + w(t) \quad (2)$$

where p is the order of the AR process, $\{\alpha_\tau\}$ the AR coefficients, and $w(t)$ denotes white noise. The source propagates to sensor j via a multi-path channel modeled as an FIR filter $C_j(z) = \sum_{l=0}^{L_j-1} c_{j,l} z^{-l}$, of *unknown* order L_j and tap coefficients $\{c_{j,l}\}$. In the presence of additive sensing noise $\bar{\epsilon}_j(t)$, the observation at sensor j is $x_j(t) = \sum_{l=0}^{L_j-1} c_{j,l}\theta(t - l) + \bar{\epsilon}_j(t)$. Since $x_j(t)$ is an ARMA process, it can be written as [10]

$$x_j(t) = -\sum_{\tau=1}^p \alpha_\tau x_j(t - \tau) + \sum_{\tau'=1}^m \beta_{\tau'} \bar{\xi}_j(t - \tau'), \quad j \in \mathcal{J} \quad (3)$$

where the moving average (MA) coefficients $\{\beta_{\tau'}\}$ and the variance of the white noise process $\bar{\xi}_j(t)$ depend on $\{c_{j,l}\}$, $\{\alpha_\tau\}$, and the variance of the noise terms $w(t)$ and $\bar{\epsilon}_j(t)$. For the purpose of determining spectral peaks, the MA term in (3) can be treated as observation noise, i.e., $\epsilon_j(t) := \sum_{\tau'=1}^m \beta_{\tau'} \bar{\xi}_j(t - \tau')$. This is important since sensors do not have to know the source-sensor channel coefficients as well as the noise

variances. The spectral content of the source can be obtained provided sensors estimate the coefficients $\{\alpha_\tau\}$, so let $\mathbf{s}_0 := [\alpha_1 \ldots \alpha_p]^T$. Note from (3) that the regressor vectors here are $\mathbf{h}_j(t) = [-x_j(t-1) \cdots - x_j(t-p)]^T$, directly from the sensor data $\{x_j(t)\}$ without the need of training/estimation.

Remark 2: The source-sensor channels may introduce deep fades at the frequencies occupied by the source. Thus, having each sensor operating on its own may lead to faulty assessments. The necessary spatial diversity to effect improved spectral estimates, can only be achieved through sensor collaboration as in the D-RLS algorithm described next.

III. DISTRIBUTED RLS ALGORITHM

In this section, we first construct the D-RLS algorithm, and then provide further insights regarding its implementation and associated communication overhead and computational complexity. The approach followed consists of two main steps: i) reformulate (1) into an equivalent separable minimization problem that is amenable to distributed implementation; and ii) rely on the AD-MoM [1, p. 253] to split (1) into simpler optimization subtasks that can be carried out locally at each sensor.

To decompose the cost function in (1), in which summands are coupled through the global variable $\mathbf{s}$, we introduce auxiliary variables $\{\mathbf{s}_j\}_{j=1}^J$ that represent local estimates of $\mathbf{s}_0$ per sensor j. These local estimates are utilized to form the convex *constrained* minimization problem:

$$\{\hat{\mathbf{s}}_j(t)\}_{j=1}^J := \arg \min_{\{\mathbf{s}_j\}_{j=1}^J} \sum_{\tau=0}^t \sum_{j=1}^J \lambda^{t-\tau} [x_j(\tau) - \mathbf{h}_j^T(\tau)\mathbf{s}_j]^2$$
$$+ J^{-1}\lambda^t \sum_{j=1}^J \mathbf{s}_j^T \boldsymbol{\Phi}_0 \mathbf{s}_j,$$
$$\text{s. t. } \mathbf{s}_j = \mathbf{s}_{j'}, \quad j \in \mathcal{J}, \ j' \in \mathcal{N}_j. \quad (4)$$

From the connectivity of the WSN, (1) and (4) are equivalent in the sense that $\hat{\mathbf{s}}_j(t) = \hat{\mathbf{s}}_{\mathrm{ewls}}(t), \forall j \in \mathcal{J}$ and $t \geq 0$; see also [9].

Algorithm Construction

In order to tackle (4) in a distributed fashion, we resort to AD-MoM to obtain an adaptive algorithm that: i) allows recursive estimation of a time-invariant parameter $\mathbf{s}_0$; and ii) can track a time-varying process $\mathbf{s}_0(t)$. To facilitate application of AD-MoM, consider the auxiliary variables $\{\mathbf{z}_j^{j'}\}_{j' \in \mathcal{N}_j}$ for $j \in \mathcal{J}$, and replace the constraints in (4) with the equivalent ones

$$\mathbf{s}_j = \mathbf{z}_j^{j'} \text{ and } \mathbf{s}_{j'} = \mathbf{z}_j^{j'}, \quad j \in \mathcal{J}, \ j' \in \mathcal{N}_j, \ j \neq j'. \quad (5)$$

Variables $\mathbf{z}_j^{j'}$ are only used to derive the local recursions but will be eventually eliminated. Next, associate Lagrange multipliers $\mathbf{v}_j^{j'}$ and $\boldsymbol{\mu}_j^{j'}$ with the constraints in (5), and form the quadratically augmented Lagrangian

$$\mathcal{L}_a[\boldsymbol{s}, \mathbf{z}, \mathbf{v}, \boldsymbol{\mu}] = \sum_{j=1}^J \sum_{\tau=1}^t \lambda^{t-\tau} [x_j(\tau) - \mathbf{h}_j^T(\tau)\mathbf{s}_j]^2 + \frac{\lambda^t}{J} \sum_{j=1}^J \mathbf{s}_j^T \boldsymbol{\Phi}_0 \mathbf{s}_j$$
$$+ \sum_{j=1}^J \sum_{j' \in \mathcal{N}_j} \left[\left(\mathbf{v}_j^{j'}\right)^T (\mathbf{s}_j - \mathbf{z}_j^{j'}) + \left(\boldsymbol{\mu}_j^{j'}\right)^T (\mathbf{s}_j - \mathbf{z}_{j'}^j) \right]$$
$$+ \frac{c}{2} \sum_{j=1}^J \sum_{j' \in \mathcal{N}_j} \left[\|\mathbf{s}_j - \mathbf{z}_j^{j'}\|^2 + \|\mathbf{s}_j - \mathbf{z}_{j'}^j\|^2 \right] \quad (6)$$

where c is positive penalty coefficient; and $\boldsymbol{s} := \{\mathbf{s}_j\}_{j=1}^J$, $\mathbf{z} := \{\mathbf{z}_j^{j'}\}_{j \in \mathcal{J}}^{j' \in \mathcal{N}_j}$ and $[\mathbf{v}, \boldsymbol{\mu}] := \{\mathbf{v}_j^{j'}, \boldsymbol{\mu}_j^{j'}\}_{j \in \mathcal{J}}^{j' \in \mathcal{N}_j}$. Now, let

$k = 0, 1, \ldots$ denote the iteration index for the recursive algorithm to be constructed in order to minimize (4) at time instant $t + 1$. The first step in the AD-MoM updates the multipliers using the gradient ascent iterations

$$
\mathbf{v}_j^{j'}(t + 1; k) = \mathbf{v}_j^{j'}(t + 1; k - 1)
$$
$$
+ c[\mathbf{s}_j(t + 1; k) - \mathbf{z}_j^{j'}(t + 1; k)] \quad (7)
$$
$$
\boldsymbol{\mu}_j^{j'}(t + 1; k) = \boldsymbol{\mu}_j^{j'}(t + 1; k - 1)
$$
$$
+ c[\mathbf{s}_j(t + 1; k) - \mathbf{z}_{j'}^{j}(t + 1; k)] \quad (8)
$$

where $j \in \mathcal{J}$, and $j' \in \mathcal{N}_j$ with $j' \neq j$. The second step entails recursions that are obtained after minimizing (6) w.r.t. $\mathbf{s}$, assuming that all other variables $\mathbf{z}(t + 1; k) := \{\mathbf{z}_j^{j'}(t + 1; k)\}_{j \in \mathcal{J}}^{j' \in \mathcal{N}_j}$ and $[\mathbf{v}(t + 1; k), \boldsymbol{\mu}(t + 1; k)] := \{[\mathbf{v}_j^{j'}(t + 1; k), \boldsymbol{\mu}_j^{j'}(t + 1; k)]\}_{j \in \mathcal{J}}^{j' \in \mathcal{N}_j}$ are fixed. The separable structure of (6) w.r.t. $\mathbf{s}_j$ leads to the J separate minimization subproblems

$$
\mathbf{s}_j(t + 1; k + 1)
$$
$$
:= \arg \min_{\mathbf{s}_j} \left[\sum_{\tau=0}^{t+1} \lambda^{t+1-\tau} [x_j(\tau) - \mathbf{h}_j^T(\tau)\mathbf{s}_j]^2 + J^{-1}\lambda^{t+1}\mathbf{s}_j^T \boldsymbol{\Phi}_0 \mathbf{s}_j \right.
$$
$$
+ \sum_{j' \in \mathcal{N}_j} [\mathbf{v}_j^{j'}(t + 1; k) + \boldsymbol{\mu}_j^{j'}(t + 1; k)]^T \mathbf{s}_j
$$
$$
\left. + \frac{c}{2} \sum_{j' \in \mathcal{N}_j} \left(\|\mathbf{s}_j - \mathbf{z}_j^{j'}(t + 1; k)\|^2 + \|\mathbf{s}_j - \mathbf{z}_{j'}^{j}(t + 1; k)\|^2 \right) \right]
$$
$$
(9)
$$

which are quadratic and whose optimal solution is available in closed-form.

The third step involves updating $\mathbf{z}_j^{j'}(t + 1; k)$. The related recursions are obtained after minimizing $\mathcal{L}_a [\mathbf{s}(t+1; k+1), \mathbf{z}, \mathbf{v}(t + 1; k), \boldsymbol{\mu}(t + 1; k)]$ w.r.t. $\mathbf{z}$, while treating $\mathbf{s}(t + 1; k + 1) := \{\mathbf{s}_j(t+1; k+1)\}_{j \in \mathcal{J}}$ and $[\mathbf{v}(t+1; k), \boldsymbol{\mu}(t + 1; k)] := \{[\mathbf{v}_j^{j'}(t + 1; k), \boldsymbol{\mu}_j^{j'}(t + 1; k)]\}_{j \in \mathcal{J}}^{j' \in \mathcal{N}_j}$ as fixed. Then, given the separable structure of the Lagrangian in (6) w.r.t. $\mathbf{z}_j^{j'}$, it follows after retaining only the $\mathbf{z}_j^{j'}$-dependent terms in (6) that

$$
\mathbf{z}_j^{j'}(t + 1; k + 1) := \arg \min_{\mathbf{z}_j^{j'}} \left[-[\mathbf{v}_j^{j'}(t + 1; k) + \boldsymbol{\mu}_{j'}^{j}(t + 1; k)]^T \mathbf{z}_j^{j'} \right.
$$
$$
\left. + \frac{c}{2} \left[\|\mathbf{s}_j(t + 1; k + 1) - \mathbf{z}_j^{j'}\|^2 + \|\mathbf{s}_{j'}(t + 1; k + 1) - \mathbf{z}_j^{j'}\|^2 \right] \right]
$$

which being linear-quadratic accepts the closed-form solution

$$
\mathbf{z}_j^{j'}(t + 1; k + 1) = \frac{1}{2c} \left[\mathbf{v}_j^{j'}(t + 1; k) + \boldsymbol{\mu}_{j'}^{j}(t + 1; k) \right]
$$
$$
+ \frac{1}{2} [\mathbf{s}_j(t + 1; k + 1) + \mathbf{s}_{j'}(t + 1; k + 1)]. \quad (10)
$$

Substituting (10) into (7) and (8), it follows that if the Lagrange multipliers are initialized such that $\mathbf{v}_j^{j'}(t + 1; 0) = -\boldsymbol{\mu}_{j'}^{j}(t + 1; 0)$, then $\mathbf{v}_j^{j'}(t + 1; k) = -\boldsymbol{\mu}_{j'}^{j}(t + 1; k)$ for all t and k, while

$$
\mathbf{v}_j^{j'}(t+1; k) = \mathbf{v}_j^{j'}(t+1; k-1) + \frac{c}{2} [\mathbf{s}_j(t+1; k) - \mathbf{s}_{j'}(t+1; k)] \quad (11)
$$

where $j \in \mathcal{J}$ and $j' \in \mathcal{N}_j$. Notice that sensor j has to store and update only $\{\mathbf{v}_j^{j'}(t + 1; k)\}_{j' \in \mathcal{N}_j}$ since $\boldsymbol{\mu}_j^{j'}$ turned out to be redundant. To obtain a recursion for $\mathbf{s}_j(t + 1; k + 1)$: i) substitute (10) into (9); ii) use the identity $\boldsymbol{\mu}_j^{j'}(t + 1; k) = -\mathbf{v}_{j'}^{j}(t + 1; k)$ to eliminate $\boldsymbol{\mu}_j^{j'}$ from (9);

and iii) apply first-order optimality conditions to the resulting quadratic cost. Then, $\mathbf{s}_j(t + 1; k + 1)$ can be obtained recursively as

$$
\mathbf{s}_j(t + 1; k + 1)
$$
$$
= \boldsymbol{\Phi}_j^{-1}(t + 1)\boldsymbol{\psi}_j(t + 1)
$$
$$
+ \frac{c}{2}\boldsymbol{\Phi}_j^{-1}(t + 1) \left(|\mathcal{N}_j|\mathbf{s}_j(t + 1; k) + \sum_{j' \in \mathcal{N}_j} \mathbf{s}_{j'}(t + 1; k) \right)
$$
$$
- \frac{1}{2}\boldsymbol{\Phi}_j^{-1}(t + 1) \sum_{j' \in \mathcal{N}_j} [\mathbf{v}_j^{j'}(t + 1; k) - \mathbf{v}_{j'}^{j}(t + 1; k)] \quad (12)
$$

where

$$
\boldsymbol{\Phi}_j(t + 1) := \sum_{\tau=0}^{t+1} \lambda^{t+1-\tau} \mathbf{h}_j(\tau)\mathbf{h}_j^T(\tau) + J^{-1}\lambda^{t+1}\boldsymbol{\Phi}_0 + c|\mathcal{N}_j|\mathbf{I}_p,
$$
$$
\boldsymbol{\psi}_j(t + 1) := \sum_{\tau=0}^{t+1} \lambda^{t+1-\tau} \mathbf{h}_j(\tau) x_j(\tau)
$$
$$
= \lambda\boldsymbol{\psi}_j(t) + \mathbf{h}_j(t + 1) x_j(t + 1).
$$

Recursions (11) and (12) constitute the D-RLS algorithm, whereby all sensors $j \in \mathcal{J}$ keep track of their local estimate $\mathbf{s}_j(t + 1; k + 1)$ and their multipliers $\{\mathbf{v}_j^{j'}(t + 1; k)\}_{j' \in \mathcal{N}_j}$, which can be arbitrarily initialized. For $\lambda = 1$, matrix $\boldsymbol{\Phi}_j^{-1}(t + 1)$ can be also recursively obtained from $\boldsymbol{\Phi}_j^{-1}(t)$ with complexity $\mathcal{O}(p^2)$ using the matrix inversion lemma; i.e.,

$$
\boldsymbol{\Phi}_j^{-1}(t + 1) = \boldsymbol{\Phi}_j^{-1}(t) - \frac{\boldsymbol{\Phi}_j^{-1}(t)\mathbf{h}_j(t + 1)\mathbf{h}_j^T(t + 1)\boldsymbol{\Phi}_j^{-1}(t)}{1 + \mathbf{h}_j^T(t + 1)\boldsymbol{\Phi}_j^{-1}(t)\mathbf{h}_j(t + 1)}.
$$

Interestingly, the first term in $\mathbf{s}_j(t + 1; k + 1)$, namely $\boldsymbol{\Phi}_j(t + 1)^{-1}\boldsymbol{\psi}_j(t + 1)$, is a regularized version of the local EWLSE per sensor j at time instant $t + 1$. The regularization is imposed by the scaled identity matrix term in $\boldsymbol{\Phi}_j(t + 1)$. Contrary to the classical RLS, see e.g., [5], it allows one to set $\boldsymbol{\Phi}_0 = \mathbf{0}$ without compromising the invertibility of $\boldsymbol{\Phi}_j(t)$. The remaining terms in (12) are responsible for fusing information from the neighborhood of sensor j, refining in that way the local estimate provided by $\boldsymbol{\Phi}_j^{-1}(t + 1)\boldsymbol{\psi}_j(t + 1)$. As promised, the variables $\mathbf{z}_j^{j'}$ have been completely eliminated from the D-RLS recursions in (11)–(12).

In order to solve (4) at time instant $t + 1$, all sensors run local consensus recursions. During the $(k + 1)$st consensus iteration, sensor j receives the local estimates $\mathbf{s}_{j'}(t + 1; k)$ from its neighbors $j' \in \mathcal{N}_j$ and updates its multipliers $\mathbf{v}_j^{j'}(t + 1; k)$ via (11). Then, sensor j receives the multipliers $\{\mathbf{v}_j^{j}(t + 1; k)\}_{j' \in \mathcal{N}_j}$ and uses them along with $\{\mathbf{s}_{j'}(t + 1; k)\}_{j' \in \mathcal{N}_j}$ to evaluate $\mathbf{s}_j(t + 1; k + 1)$ via (12). This way, recursions (11)–(12) minimize (4) asymptotically. Specifically, it follows that:

Proposition 1: *For arbitrarily initialized* $\{\mathbf{v}_j^{j'}(t; 0)\}_{j \in \mathcal{J}}^{j' \in \mathcal{N}_j}$, $\mathbf{s}_j(t; 0)$ *and any* $c > 0$; *the local estimates* $\mathbf{s}_j(t; k)$ *reach consensus as* $k \to \infty$; *i.e.,*

$$
\lim_{k \to \infty} \mathbf{s}_j(t; k) = \hat{\mathbf{s}}_{\text{ewls}}(t), \text{ for all } j \in \mathcal{J}.
$$

Proof: For any t, the arguments in [9, Appendix B] apply directly to obtain the convergence claim. ∎

Thus, D-RLS recursions are able to attain the EWLSE at each time instant t as long as the number of consensus iterations grows. For a time-invariant setup, running many consensus iterations, i.e., $k \gg 1$

would not be a problem, though this is not the case when the sensors track a time-varying process $\mathbf{s}_0(t)$. One way to enable D-RLS operation in nonstationary settings, is to apply one consensus iteration per time instant t. In this case, $k = t$ and recursions (11)–(12) simplify to

$$
\begin{aligned}
\mathbf{v}_j^{j'}(t) &= \mathbf{v}_j^{j'}(t-1) + \tfrac{c}{2}[\mathbf{s}_j(t) - (\mathbf{s}_{j'}(t) + \boldsymbol{\eta}_j^{j'}(t))], \qquad (13)\\
\mathbf{s}_j(t+1) &= \boldsymbol{\Phi}_j^{-1}(t+1)\boldsymbol{\psi}_j(t+1)\\
&\quad + \tfrac{c}{2}\boldsymbol{\Phi}_j^{-1}(t+1)\left(|\mathcal{N}_j|\mathbf{s}_j(t) + \sum_{j'\in\mathcal{N}_j}(\mathbf{s}_{j'}(t)+\boldsymbol{\eta}_j^{j'}(t))\right)\\
&\quad - \tfrac{1}{2}\boldsymbol{\Phi}_j^{-1}(t+1)\sum_{j'\in\mathcal{N}_j}[\mathbf{v}_j^{j'}(t) - (\mathbf{v}_{j'}^j(t)+\bar{\boldsymbol{\eta}}_j^{j'}(t))] \qquad (14)
\end{aligned}
$$

where $\boldsymbol{\eta}_j^{j'}(t)$ and $\bar{\boldsymbol{\eta}}_j^{j'}(t)$ denote the additive communication noise present in the reception of $\mathbf{s}_j(t)$ and $\mathbf{v}_{j'}^j(t)$ at sensor j, respectively. In detail, during time instant $t+1$ sensor j receives the local estimates $\{\mathbf{s}_{j'}(t) + \boldsymbol{\eta}_j^{j'}(t)\}_{j'\in\mathcal{N}_j}$ and plugs them into (13) to evaluate $\mathbf{v}_j^{j'}(t)$ for $j' \in \mathcal{N}_j$. Then, it receives $\mathbf{v}_{j'}^j(t) + \bar{\boldsymbol{\eta}}_j^{j'}(t)$ from its neighbors $j' \in \mathcal{N}_j$, which are used together with $\{\mathbf{s}_{j'}(t) + \boldsymbol{\eta}_j^{j'}(t)\}_{j'\in\mathcal{N}_j}$ and the new observation data within $\boldsymbol{\Phi}_j^{-1}(t+1)\boldsymbol{\psi}_j(t+1)$ to obtain $\mathbf{s}_j(t+1)$ via (14). Recursions (13)–(14) constitute a single-time (ST-) scale version of D-RLS, abbreviated as STD-RLS. Note also that there is no need for a common penalty coefficient c across sensors; that is, each sensor can use its own local penalty coefficient $c_j > 0$ allowing increased flexibility to attain potentially higher convergence rates.

Remark 3: A similar consensus-based RLS algorithm was put forth in the conference precursor of this paper [7]. To enable task parallelization via AD-MoM while ensuring that estimates agree across the whole WSN, the approach in [7] judiciously reformulates (1) by relying on the so called bridge sensor subset. Not only setting up—but readjusting the bridge sensor set, e.g., when sensors inevitably fail in battery-limited WSN deployments—requires additional coordination among sensors with an associated communication overhead. Compared to [7], the approach followed here does not require such a bridge sensor set, and in this sense, it offers a fully distributed, robust, and resource efficient RLS-type algorithm for use in ad hoc WSNs.

A. Communication and Computational Costs

Next, we analyze the communication and computational costs associated with D-RLS, and compare them with those incurred by existing approaches. Per D-RLS iteration each sensor transmits $p(|\mathcal{N}_j| + 1)$ scalars corresponding to the multipliers $\{\mathbf{v}_j^{j'}\}_{j'\in\mathcal{N}_j}$, and the local estimate $\mathbf{s}_j$. In diffusion RLS [2], each sensor transmits $2p + 1$ scalars per iteration. However, when considering the reception cost it can be seen that while in D-RLS each sensor receives $2|\mathcal{N}_j|p$ scalars per recursion, in diffusion RLS the number of received scalars increases to $|\mathcal{N}_j|(2p + 1)$ per iteration. Even though the transmission cost is arguably greater than the one related to reception, it will be corroborated via numerical examples that the higher transmission cost in D-RLS pays off in improved convergence rates and robustness in the presence of communication noise.

The communication cost for I-RLS in [6] is $\mathcal{O}(p^2)$, since each sensor has to transmits to its successor in the Hamiltonian cycle a $p \times p$ covariance matrix; similar complexity is incurred by the scheme in [11]. A low communication cost I-RLS is also proposed in [6] in which each sensor within the cycle transmits and receives p scalars per iteration, though the challenges related to I-RLS remain as the WSN scales.

Interestingly, when communication noise is not present as in the scenarios considered in [2], [6], [11], D-RLS can be modified such that its corresponding communication complexity becomes lower than the one incurred by diffusion RLS. Specifically, note that if the multipliers $\mathbf{v}_j^{j'}$ are initialized such that $\mathbf{v}_j^{j'}(t;0) = -\mathbf{v}_{j'}^j(t;0)$, then in the absence of

noise $\mathbf{v}_j^{j'}(t;k) = -\mathbf{v}_{j'}^j(t;k)$ for all k and t [cf. (7)]. Similarly, for the STD-RLS it follows that if $\mathbf{v}_j^{j'}(0) = -\mathbf{v}_{j'}^j(0)$, and noise is not present then $\mathbf{v}_j^{j'}(t) = -\mathbf{v}_{j'}^j(t)$ for all t. Taking into account this equality, and setting $\bar{\boldsymbol{\eta}}_j^{j'}(t) = \boldsymbol{\eta}_j^{j'}(t) = \mathbf{0}$, the recursion for $\mathbf{s}_j(t+1)$ in STD-RLS is rewritten as [the same can be done with (12)]

$$
\begin{aligned}
\mathbf{s}_j(t+1) &= \boldsymbol{\Phi}_j^{-1}(t+1)\boldsymbol{\psi}_j(t+1) - \boldsymbol{\Phi}_j^{-1}(t+1)\sum_{j'\in\mathcal{N}_j}\mathbf{v}_j^{j'}(t)\\
&\quad + \tfrac{c}{2}\boldsymbol{\Phi}_j^{-1}(t+1)\left(|\mathcal{N}_j|\mathbf{s}_j(t) + \sum_{j'\in\mathcal{N}_j}\mathbf{s}_{j'}(t)\right). \quad (15)
\end{aligned}
$$

The second summand on the right-hand side of (15) incorporates only local multipliers stored at sensor j. Thus, each sensor does not exchange multipliers with its neighbors to update $\mathbf{s}_j(t + 1)$. When using the modified STD-RLS comprising recursions (13) and (15), each sensor transmits p scalars and receives $|\mathcal{N}_j|p$ scalars per iteration. Clearly, the communication overhead is smaller than the one associated with diffusion RLS. However, as in diffusion RLS the low transmission cost is counterbalanced by the lack of resilience in the presence of communication noise.

Next, we focus on the computational complexity involved in implementing (13)–(14). Updating the multipliers incurs complexity in the order of $\mathcal{O}(|\mathcal{N}_j|p)$. In determining $\mathbf{s}_j(t + 1)$, the dominating cost arises from calculating $\boldsymbol{\Phi}_j^{-1}(t + 1)$. Recall that when $\lambda = 1$, matrix $\boldsymbol{\Phi}_j^{-1}(t+1)$ can be computed recursively with a complexity of $\mathcal{O}(p^2)$. If $\lambda < 1$, then the complexity for determining $\boldsymbol{\Phi}_j^{-1}(t + 1)$ is $\mathcal{O}(p^3)$. In diffusion RLS, the computational complexity is also dominated by the cost of recursively updating the inverse of the regression covariance matrix, and is of order $\mathcal{O}(|\mathcal{N}_j|p^2)$. Thus, for $\lambda = 1$ the computational complexity per iteration is smaller than the one in diffusion RLS. For $\lambda < 1$, the way D-RLS and diffusion RLS compare in terms of computational complexity depends on the relative size of $\{|\mathcal{N}_j|\}_{j=1}^J$ and p. Specifically, if $p < |\mathcal{N}_j|$ (e.g., in localization applications where $p \leq 3$), then D-RLS incurs smaller complexity. While, if $p > |\mathcal{N}_j|$ diffusion RLS is less complex computationally.

Remark 4: Despite the fact that the computational complexity of (ST-)D-RLS depends on the estimation setting, the novel algorithm enjoys communication noise resilience, not present in existing alternatives. This property makes (ST-)D-RLS a viable candidate for estimation/tracking in WSNs. It is also an expected feature since both algorithms rely on the AD-MoM, which exhibits robustness in the presence of communication noise [9].

These claims are also supported by simulations in the next section, which corroborate the advantages of the proposed approach both in terms of estimation performance as well as noise resilience.

IV. NUMERICAL TESTS

Here we test the novel D-RLS and STD-RLS algorithms in the spectral application setting described in Section II-B, conducting several performance comparisons with: i) I-RLS [6]; ii) diffusion RLS using Metropolis weights [2]; iii) local (L-) RLS, whereby each sensor runs an independent RLS algorithm solely based on its own data (no inter-sensor communications); and iv) D-LMS with step-size $\mu = 10^{-2}$ [8]. For $J = 30$ sensors, an ad hoc WSN is generated by using the random geometric graph model in $[0, 1]^2$, with communication range $r = 0.6$; see Fig. 1. For the examples with noisy links, additive white Gaussian noise (AWGN) with variance $\sigma_\eta^2 = 10^{-1}$ is added at the receiving end. The source $\theta(t)$ is an AR (4) process with coefficients $\mathbf{s}_0 = [-0.31, 1.14, 0.28, 0.22]^T$ and driving noise variance $\sigma_w^2 = 10^{-2}$, which yields a spectrum with a single peak at $\omega = \pi/2$. The source-sensor channels have order $L_j = 2$, and the channels to the sensors 3, 7, 15, and 27 vanish at the frequency $\omega = \pi/2$. The observation
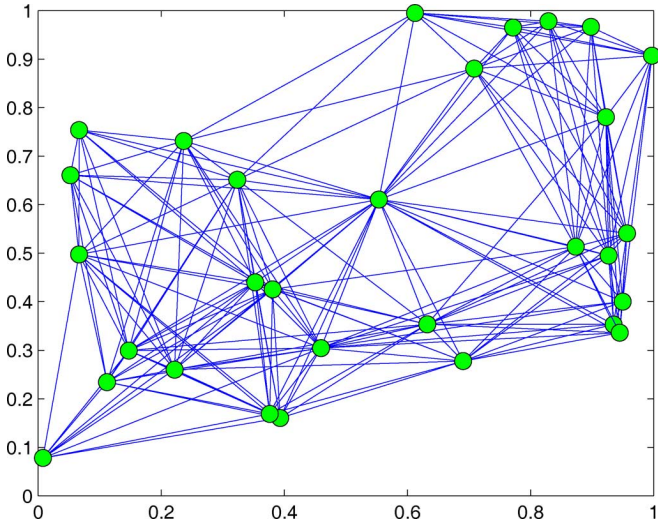
Fig. 1. Ad hoc WSN with $J = 30$ sensors.

Fig. 2. Global network performance in a distributed power spectrum estimation task: (top) MSE (learning curve); (bottom) MSD.

AWGN has a spatial variance profile $\sigma_{\epsilon_j}^2 = \alpha_j \times 10^{-4}$, where the co-efficients $\alpha_j \sim \mathcal{U}[0, 1]$ (uniformly distributed) are i.i.d. across sensors.

Thirty consensus steps are ran per acquired observation in D-RLS, in order to ensure a fair comparison with I-RLS in terms of processing delay. The delay is due to the estimation cycle over all $J = 30$ sensors, that should be completed before new information can be incorporated. With $\lambda = 1$ and $c_j = 7/|\mathcal{N}_j|$ in both D-RLS and STD-RLS, Fig. 2 (top) compares the global MSE evolution (learning curve) obtained as $J^{-1} \sum_{j=1}^{J} E[(x_j(t) - \mathbf{h}_j^T(t)\mathbf{s}_j(t-1))^2]$, whereas the expectation is approximated by averaging over 250 realizations of the experiment. Similar curves are shown in Fig. 2 (bottom), in this case for the global mean-square deviation (MSD) metric given by $J^{-1} \sum_{j=1}^{J} E[\|\mathbf{s}_j(t) - \mathbf{s}_0\|^2]$. In the absence of communication noise, the cost-effective version of STD-RLS is implemented via recursions (13) and (15). I-RLS and D-RLS behave similarly providing a performance benchmark, while D-LMS—a first-order method—converges much slower than all distributed RLS schemes. STD-RLS outperforms diffusion RLS in terms of convergence rate, and most importantly, it does not suffer from the catastrophic noise accumulation exhibited by diffusion RLS when the links are not ideal.

With regards to local performance in steady-state, for $\lambda = 0.9$ we illustrate in Fig. 3 (top) the figures of merit which are customary in the adaptive literature [2], [5], [8]: i) MSE $E[(x_j(t) - \mathbf{h}_j^T(t)\mathbf{s}_j(t - 1))^2]$; ii) excess-MSE (EMSE) $E[(\mathbf{h}_j^T(t)(\mathbf{s}_j(t-1) - \mathbf{s}_0))^2]$; and MSD $E[\|\mathbf{s}_j(t) - \mathbf{s}_0\|^2]$. With reference to Remark 2, it is apparent that a scheme devoid of sensor collaboration such as L-LMS, fails to obtain satisfactory estimates at the sensors affected by the channel fades. On the other hand, STD-RLS exploits the available spatial diversity to attain improved estimation performance, see, e.g., the local MSD curves.

Next, we illustrate the capabilities of STD-RLS when it comes to tracking a time varying parameter $\mathbf{s}_0(t)$. For $p = 6$ and for the same WSN setup, we simulate a large amplitude slowly time-varying process $\mathbf{s}_0(t) = (1 - \rho_1)\mathbf{s}_0(t - 1) + \boldsymbol{\nu}(t)$ with $\rho_1 = 0.9$ and $\boldsymbol{\nu}(t) \sim \mathcal{N}(\mathbf{0}, 10^{-2}\mathbf{I}_6)$ (multivariate normal distribution). A linear model is adopted for the sensor observations, i.e., $x_j(t) = \mathbf{h}_j^T(t)\mathbf{s}_0(t) + \zeta_j(t)$ with $\zeta_j(t) \sim \mathcal{N}(0, 10^{-4})$ for all $j \in \mathcal{J}$. Regressors are temporally correlated, as $\mathbf{h}_j(t) = [h_j(t) \dots h_j(t-5)]^T$ with entries which evolve according to $h_j(t) = (1 - \rho_2)\beta_j h_j(t - 1) + \sqrt{\rho_2}\nu_j(t)$. We choose $\rho_2 = 0.7$, the $\beta_j \sim \mathcal{U}[0, 1]$ are i.i.d. in space, and the driving white noise $\nu_j(t) \sim \mathcal{U}[-\sqrt{3}\sigma_{\nu_j}, \sqrt{3}\sigma_{\nu_j}]$ has a spatial variance profile given by $\sigma_{\nu_j}^2 = 10^{-1}\gamma_j$, with $\gamma_j \sim \mathcal{U}[0, 1]$ and i.i.d. For $\lambda = 0.5$ and $c_j = 7/|\mathcal{N}_j|$, Fig. 3 (bottom) depicts the second entry of $\mathbf{s}_0(t)$ as
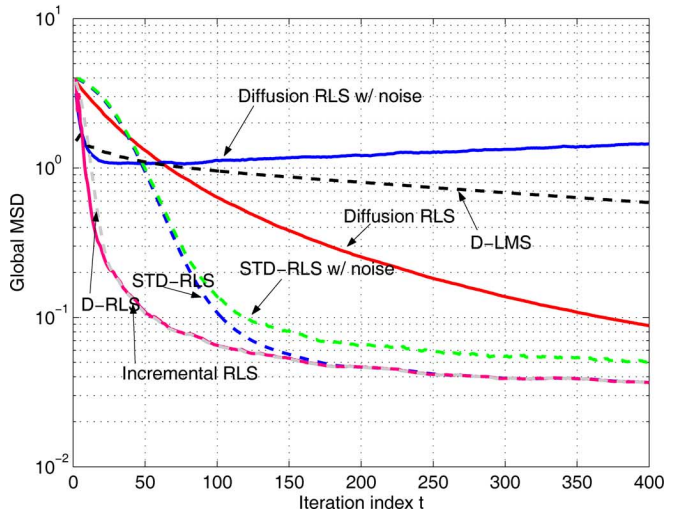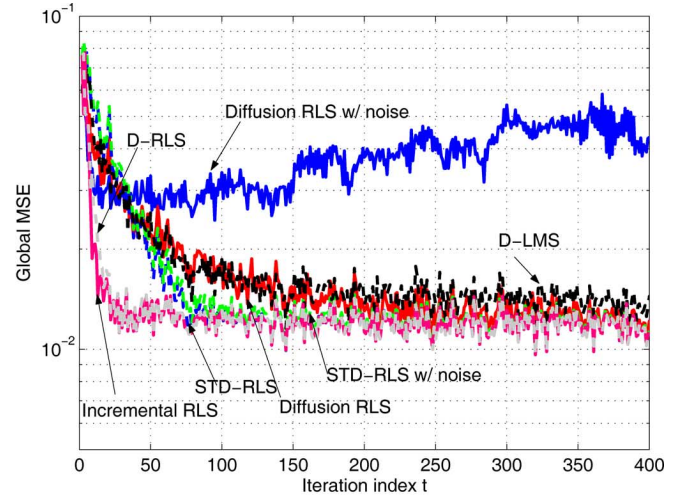
well as the corresponding estimate for a representative sensor closely tracking the true variations. In the presence of communication noise, the larger estimate fluctuations are a direct manifestation of the (expected) increased MSE, as evidenced by the learning curves in Fig. 3 (bottom).

The scheme in [11] has not been included in the numerical comparisons because a complete data model is not available for the power spectrum estimation problem. Specifically, the variance of the aggregate observation noise term $\epsilon_j(t)$ is unknown (cf. Section II-A). Further, the algorithm in [11] is incapable of tracking $\mathbf{s}_0(t)$ due to its diminishing step-size.

V. CONCLUDING REMARKS

We developed a distributed RLS-like algorithm for adaptive estimation/tracking using WSNs in which sensors communicate via single-hop noisy links. The approach adopted involves i) reformulating in a separable way the exponentially weighed least-squares cost involved in the classical RLS algorithm; and ii) applying the AD-MoM scheme to minimize this separable cost in a distributed fashion. The resulting algorithm entails only local computational tasks across sensors that simply exchange messages with single-hop neighbors only.

In order to accommodate nonstationary applications, STD-RLS was derived from D-RLS entailing a single consensus recursion per time
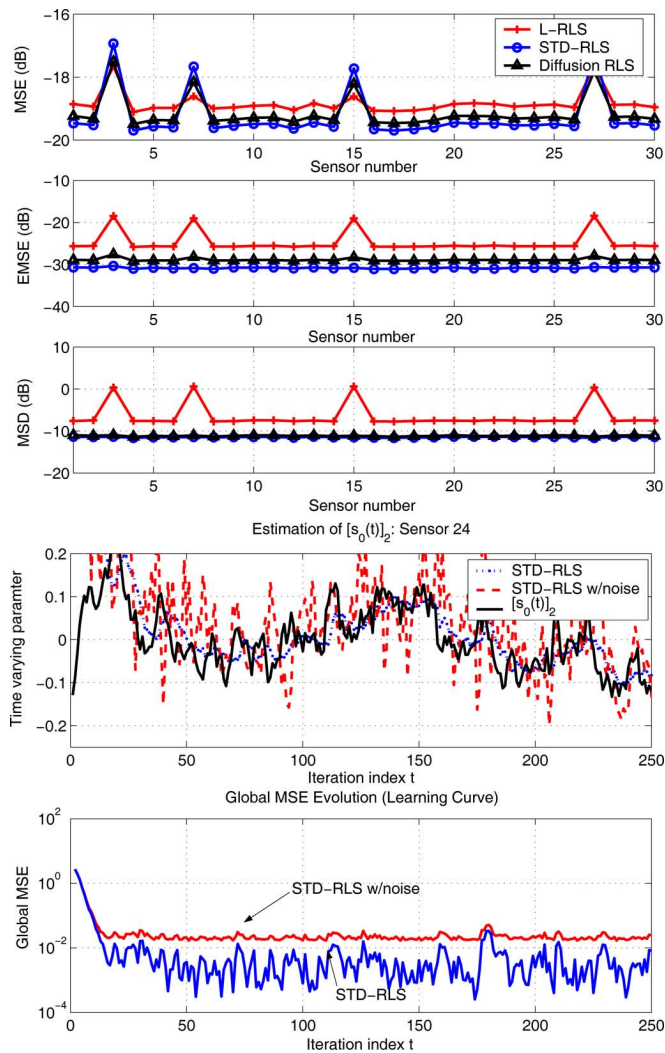
Fig. 3. (top) Local (per-sensor) performance in a distributed power spectrum estimation task; (bottom) Tracking with STD-RLS.

instant t. Different from existing approaches, (ST)D-RLS exhibits robustness to inter-sensor communication noise at the expense of a slightly higher communication complexity. When noise is not an issue, (ST)D-RLS can be modified to lower communication overhead relative to adaptive diffusion estimation schemes. Numerical examples demonstrate the noise robustness of (ST)D-RLS, as well as its convergence and tracking capabilities in comparison with existing alternatives.[1]

REFERENCES

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

[2] F. S. Cattivelli, C. G. Lopes, and A. H. Sayed, "Diffusion recursive least-squares for distributed estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1865–1877, May 2008.

[3] 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, Jul. 2008.

[4] C. H. Papadimitriou, *Computational Complexity*. Reading, MA: Addison-Wesley, 1993.

[5] A. H. Sayed, *Fundamentals of Adaptive Filtering*. New York: Wiley, 2003.

[6] A. H. Sayed and C. G. Lopes, "Distributed recursive least-squares over adaptive networks," in *Proc. 40th Asilomar Conf. On Signals, Systems and Computers*, Pacific Grove, CA, Oct./Nov. 2006, pp. 233–237.

[7] I. D. Schizas, G. Mateos, and G. B. Giannakis, "Consensus-based distributed recursive least-squares estimation using ad hoc wireless sensor networks," in *Proc. 41st Asilomar Conf. Signals, Systems, Computers*, Pacific Grove, CA, Nov. 2007, pp. 386–390.

[8] I. D. Schizas, G. Mateos, and G. B. Giannakis, "Distributed LMS for consesus-based in-network adaptive processing," *IEEE Trans. Signal Process.*, vol. 57, no. 6, pp. 2365–2382, Jun. 2009.

[9] 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. Signal Process.*, vol. 56, no. 1, pp. 350–364, Jan. 2008.

[10] P. Stoica and R. Moses, *Spectral Analysis of Signals*. Englewood Cliffs, NJ: Prentice-Hall, 2005.

[11] L. Xiao, S. Boyd, and S. Lall, "A space-time diffusion scheme for peer-to-peer least-squares estimation," in *Proc. Int. Conf. Info. Proc. Sensor Networks*, Nashville, TN, 2006, pp. 168–176.

Joint Transmitter/Receiver I/Q Imbalance Compensation for Direct Conversion OFDM in Packet-Switched Multipath Environments

Jeffrey Feigin and David Brady

Abstract—This correspondence presents an algorithm for compensation of I/Q imbalance for a direct-conversion packet-switched orthogonal frequency-division-multiplexing (OFDM) communications system, which accounts for transmitter I/Q imbalance, receiver I/Q imbalance, phase/frequency error, and dispersive multipath fading. The proposed estimation algorithm is then presented, which operates within the framework of existing multiuser OFDM radio standards (802.11a). It is shown that this algorithm accurately estimates and corrects transceiver I/Q imbalance on a packet-by-packet basis, all within the receiver's digital baseband.

Index Terms—Direct conversion, IEEE802.11, IEEE802.16, I/Q error, I/Q imbalance, OFDM, WIMAX, WLAN, zero-IF.

I. INTRODUCTION

Orthogonal frequency-division-multiplexing (OFDM) modulation enables low cost and current consumption while supporting high spectral efficiency through dispersive channels, and it is the predominant modulation format for some wireless communications systems. Additionally, a direct-conversion radio architecture provides the potential for excellent current consumption, size, and radio performance, and it inherently allows a great degree of channel bandwidth flexibility. However, uncalibrated direct-conversion transceivers suffer from I/Q