Consensus in *Ad Hoc* WSNs With Noisy Links—Part I: Distributed Estimation of Deterministic Signals

Ioannis D. Schizas, Alejandro Ribeiro, and Georgios B. Giannakis, *Fellow, IEEE*

***Abstract*—We deal with distributed estimation of deterministic vector parameters using *ad hoc* wireless sensor networks (WSNs). We cast the decentralized estimation problem as the solution of multiple constrained convex optimization subproblems. Using the method of multipliers in conjunction with a block coordinate descent approach we demonstrate how the resultant algorithm can be decomposed into a set of simpler tasks suitable for distributed implementation. Different from existing alternatives, our approach does not require the centralized estimator to be expressible in a separable closed form in terms of averages, thus allowing for decentralized computation even of nonlinear estimators, including maximum likelihood estimators (MLE) in nonlinear and non-Gaussian data models. We prove that these algorithms have guaranteed convergence to the desired estimator when the sensor links are assumed ideal. Furthermore, our decentralized algorithms exhibit resilience in the presence of receiver and/or quantization noise. In particular, we introduce a decentralized scheme for least-squares and best linear unbiased estimation (BLUE) and establish its convergence in the presence of communication noise. Our algorithms also exhibit potential for higher convergence rate with respect to existing schemes. Corroborating simulations demonstrate the merits of the novel distributed estimation algorithms.**

***Index Terms*—Distributed estimation, nonlinear optimization, wireless sensor networks (WSNs).**

I. Introduction

EVEN though the gamut of wireless sensor network (WSN)-driven applications is yet to be fully delineated, it is clear that the design of WSNs must be task-specific and adhering to stringent power and bandwidth constraints. A recently popular application of WSNs is decentralized estimation of unknown deterministic signal vectors using discrete-time samples collected across sensors. Fusion center (FC) based WSNs can perform decentralized estimation [17], but have limitations arising due to: i) the high transmission power required at each sensor to transmit its local information to the FC, that is proportional to the covered geographic area; and ii) lack of robustness in case of FC failures. These limitations are not encountered with *ad hoc* WSNs whereby each sensor communicates only with its neighbors, and the estimation task can be performed in a totally distributed fashion. Decentralized estimation algorithms for *ad hoc* WSNs i) guarantee that sensors obtain the desired estimates; ii) rely only on single-hop communications; and, iii) exhibit resilience in the presence of nonideal channel links among sensors.

Decentralized estimation using *ad hoc* WSNs is based on successive refinements of local estimates maintained at individual sensors. In a nutshell, each iteration of the algorithm comprises a communication step where the sensors interchange information with their neighbors, and an update step where each sensor uses this information to refine its local estimate. In this context, estimation of deterministic parameters in linear data models, via decentralized computation of the BLUE or the sample average estimator, was considered in [8], [18], [13], and [20] using the notion of consensus averaging. The sample mean estimator was formulated in [11] as an optimization problem, and was solved in a distributed fashion using dual decomposition techniques; see also [16] and [9] where consensus averaging was used for estimation of time-varying signals. Decentralized estimation of Gaussian random parameters was reported in [4] for stationary environments, while the dynamic case was considered in [15]. Recently, decentralized estimation of random signals in arbitrary nonlinear and non-Gaussian setups was considered in [14], while distributed estimation of stationary Markov random fields was pursued in [5].

Consensus averaging schemes are challenged by the presence of noise (nonideal sensor links), exhibiting a statistical behavior similar to that of a random walk, and eventually diverging [19]. An alternative for deterministic decentralized estimation in linear-Gaussian data models uses the notion of nonlinear mutually coupled oscillators [1], [10], whereby each sensor is viewed as an oscillator which through mutual coupling with its neighbors reaches the BLUE at its steady state. Interestingly, simulations in [1] advocate that iterations in the method of coupled oscillators exhibit noise robustness, but convergence has not been established analytically. Both consensus averaging in [18] and [20], as well as the coupled oscillators in [1], are somewhat limited in scope, in the sense that they require the desired estimator to be known in closed form as a properly defined function of averages.

Here, we focus on decentralized estimation of deterministic parameter vectors in general (possibly nonlinear and/or non-Gaussian) data models. Both MLE and BLUE schemes are considered. Our novel approach formulates the desired estimator as the solution of convex minimization subproblems that exhibit a separable structure and are thus amenable to distributed implementation. Different from [1] and [20], our framework leads to decentralized estimation algorithms even when the desired estimator in not available in closed form, as is frequently the case with MLE. We further prove that the resultant algorithms exhibit noise robustness in all cases. Specifically for the BLUE, our convergence analysis establishes that it has bounded steady-state noise covariance matrix. Finally, our algorithms are more flexible than those in [20] and [1] to tradeoff steady-state error for faster convergence.

After stating the problem in Section II, we proceed to view MLE as the optimal solution of a separable constrained convex minimization problem in Section III. We utilize the alternating-direction method of multipliers to find the MLE optimal

Manuscript received October 17, 2006; revised March 27, 2007. Work in this paper was supported by the USDoD ARO Grant W911NF-05-1-0283; and also through collaborative participation in the C&N Consortium sponsored by the U.S. ARL under the CTA Program, Cooperative Agreement DAAD19-01-2-0011. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Aleksandar Dogandzic.

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

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

Digital Object Identifier 10.1109/TSP.2007.906734

solution as the minimum of an appropriately defined augmented Lagrangian function. To this end we decompose the Lagrangian minimization into simple separable tasks (Section III-A). Convergence of the local estimate, to the centralized MLE is readily guaranteed for ideal channel links. In Sections III-C and III-D we provide motivating MLE paradigms based on unquantized or quantized observations [12]. In Section IV we consider distributed linear estimation using the BLUE which is appealing when computational simplicity is at a premium. Through the alternating direction multipliers method we develop a distributed (D)-BLUE algorithm, having similar features to the decentralized MLE. Interestingly, in Section V after applying appropriate linear transformations to D-BLUE we arrive at a decentralized scheme that exhibits improved resilience in the presence of noise (Section V-C). This algorithm has guaranteed convergence to the BLUE for ideal channel links, while its steady-state error covariance is bounded for noisy links. Simulations in Section V-E demonstrate the merits of our algorithms with respect to existing alternatives. We conclude the paper in Section VI.

II. Preliminaries and Problem Statement

Consider an *ad hoc* WSN with J sensors. We allow single-hop communications only, so that the jth sensor communicates solely with nodes i in its neighborhood $\mathcal{N}_j \subseteq [1, J]$. Sensor 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 communication links; see Fig. 1 (left). The connectivity information is summarized in the so called adjacency matrix $\mathbf{E} \in \mathbb{R}^{J \times J}$ for which $\mathbf{E}_{ji} = 1$ if $i \in \mathcal{N}_j$, while $\mathbf{E}_{ji} = 0$ if $i \notin \mathcal{N}_j$. Since $i \in \mathcal{N}_j$ if and only if $j \in \mathcal{N}_i$, the adjacency matrix is symmetric; i.e., $\mathbf{E} = \mathbf{E}^T$ (^{T} stands for transposition).

The WSN is deployed to estimate a $p \times 1$ deterministic unknown parameter vector $\mathbf{s}$ based on distributed random observations $\left\{\mathbf{x}_j \in \mathbb{R}^{L_j \times 1}\right\}_{j=1}^{J}$. The $\mathbf{x}_j$ observation is taken at the jth sensor and has probability density function (pdf) $p_j(\mathbf{x}_j; \mathbf{s})$. We further assume that observations are independent across sensors. If $p_j(\mathbf{x}_j; \mathbf{s})$ is known, the maximum likelihood estimator (MLE) is given by ($\ln$ denotes natural logarithm)

$$\hat{\mathbf{s}}_{ML} := \arg \min_{\mathbf{s} \in \mathbb{R}^{p \times 1}} -\sum_{j=1}^{J} \ln[p_j(\mathbf{x}_j; \mathbf{s})]. \tag{1}$$

Another estimation scenario arises when the observations adhere to a model for which $E[\mathbf{x}_j] = \mathbf{H}_j \mathbf{s}$ but, different from (1) only the covariance matrix $\mathbf{\Sigma}_{x_j x_j} := E[(\mathbf{x}_j - E[\mathbf{x}_j])(\mathbf{x}_j - E[\mathbf{x}_j])^T]$, and the matrices $\mathbf{H}_j$ are known per sensor. This setup arises frequently in, e.g., signal amplitude estimation, and includes as a special case the popular linear model $\mathbf{x}_j = \mathbf{H}_j \mathbf{s} + \mathbf{n}_j$[7]. A pertinent approach in this scenario where the sensor data pdf is unknown, is to form the BLUE which for zero-mean uncorrelated sensor observations is [7]

$$\hat{\mathbf{s}}_{BL} := \left(\sum_{j=1}^{J} \mathbf{H}_j^T \mathbf{\Sigma}_{x_j x_j}^{-1} \mathbf{H}_j \right)^{-1} \sum_{j=1}^{J} \mathbf{H}_j^T \mathbf{\Sigma}_{x_j x_j}^{-1} \mathbf{x}_j. \tag{2}$$

Both (1) and (2) will be considered. In particular, we will develop iterative algorithms based on communication with one-hop neighbors that generate (local) time iterates $\mathbf{s}_j(k)$ so that:

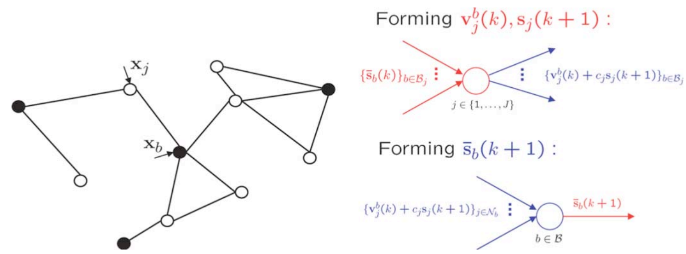

Fig. 1. (left) An *ad hoc* wireless sensor network. (right) Implementation of the D-MLE.

(s1) If $p_j(\mathbf{x}_j; \mathbf{s})$ is known only at the jth sensor, the local iterates converge as $k \to \infty$ to the global MLE, i.e., $\lim_{k\to\infty} \mathbf{s}_j(k) = \hat{\mathbf{s}}_{ML}$, with $\hat{\mathbf{s}}_{ML}$ given by (1).

(s2) If $\mathbf{\Sigma}_{x_j x_j}$ and $\mathbf{H}_j$ are known only at the jth sensor and the block matrix $\mathbf{H} := \left[\mathbf{H}_1^T \ldots \mathbf{H}_J^T\right]^T$ has full column rank, then $\lim_{k\to\infty} \mathbf{s}_j(k) = \hat{\mathbf{s}}_{BL}$, with $\hat{\mathbf{s}}_{BL}$ given by (2).

The decentralized algorithm developed under scenario (s1) is attractive for ML estimation in nonlinear data models. The linear estimator considered in (s2) is encountered in many cases of practical interest. The BLUE is generally outperformed by the MLE but its separate treatment is well motivated because it incurs lower computational complexity and remains applicable even for cases that MLE is not; e.g., when the data pdf is unknown but $\{\mathbf{H}_j\}_{j=1}^{J}$ and $\{\mathbf{\Sigma}_{x_j x_j}\}_{j=1}^{J}$ are known. Clearly, if $\mathbf{x}_j$ adheres to a linear model $\mathbf{x}_j = \mathbf{H}_j \mathbf{s} + \mathbf{n}_j$ and $\mathbf{n}_j$ is Gaussian distributed $\forall\, j$, then $\hat{\mathbf{s}}_{ML} = \hat{\mathbf{s}}_{BL}$ and consequently (s1) coincides with (s2).

Local iterates $\mathbf{s}_j(k)$ will turn out to exhibit resilience to communication noise. To describe the noisy model, let $\mathbf{t}_j^i(k) \in \mathbb{R}^{p \times 1}$ represent[1] a vector transmitted from the jth to the ith sensor at time slot k. The corresponding vector $\mathbf{r}_i^j(k) \in \mathbb{R}^{p \times 1}$ received by the ith sensor is

$$\mathbf{r}_i^j(k) = \mathbf{t}_j^i(k) + \boldsymbol{\eta}_i^j(k) \tag{3}$$

where $\boldsymbol{\eta}_i^j(k) \in \mathbb{R}^{p \times 1}$ denotes zero-mean additive noise at sensor i. Vector $\boldsymbol{\eta}_i^j(k)$ is assumed uncorrelated across sensors and time with covariance matrix $\mathbf{\Sigma}_{\eta_i \eta_i} := \mathrm{E}\left[\boldsymbol{\eta}_i^j(k) \boldsymbol{\eta}_i^{j^T}(k)\right]$. Communication noise in (3) is not necessarily Gaussian allowing us to cover:

(n1) Analog communication in the presence of additive Gaussian noise (AGN) in which case $\boldsymbol{\eta}_i^j(k)$ is normal.

(n2) Digital communication whereby each entry of $\mathbf{t}_j^i(k)$ is quantized at the jth sensor before transmission. If an m_j-bit quantizer is used with dynamic range $[-Q_j, Q_j]$, then $\boldsymbol{\eta}_i^j(k)$ is uniformly distributed in the interval $\left[-Q_j/2^{m_j}, Q_j/2^{m_j}\right]$, with covariance matrix $\mathbf{\Sigma}_{\eta_j \eta_j} = \sigma_j^2 \mathbf{I}_p$, where $\sigma_j^2 = 2^{-2m_j} Q_j^2 / 3$ and $\mathbf{I}_p$ denotes the $p \times p$ identity matrix.

Noise models (n1) and (n2) will be used in the simulations. But the ensuing robustness claims do not depend on the noise pdf. An algorithm for scenario (s1) with ideal communication links will be developed in Section III; we will also argue resilience of this algorithm to noisy communication channels as in (n1)–(n2) in Section III-B Scenario (s2) is analyzed in Section IV for noiseless links and in Section V-B for analog or digital noisy links. Throughout, we further assume that:

[1]Throughout the paper, subscripts denote the sensor at which variables are "controlled" (e.g., computed at and/or transmitted to neighbor sensors), while superscripts specify the sensor to which the variable is communicated.

(a1) the communication graph is connected; i.e., there exists a path connecting any two sensors;
(a2) the pdfs $p_j(\mathbf{x}_j; \mathbf{s})$ are log-concave with respect to the unknown parameter vector $\mathbf{s}$.

Similar to [8], [18], [20], and [1], network connectivity in (a1) ensures utilization of all observation vectors by the decentralized algorithms. The log-concavity in (a2) guarantees global identifiability (uniqueness) of the centralized ML estimator and is satisfied by a number of unimodal pdfs encountered in practice see, e.g., [12] and Section III-D. Note that unlike [11] there is no need to assume strict concavity of each summand involving $p_j(\mathbf{x}_j; \mathbf{s})$; i.e., no local identifiability is required. We close this section with a pertinent remark.

Remark 1: Different from, e.g., [3] and [12] that rely on a fusion center, *ad hoc* WSN based estimators may consume less power and are less prone to failures. Unlike existing approaches based on *ad hoc* WSNs, e.g., [20], the formulation here accounts for receiver and/or quantization noise effects.

III. Distributed MLE

In this section we consider decentralized estimation of $\hat{\mathbf{s}}_{ML}$ in (s1), under (a1) and (a2). Our approach is to rewrite the estimator in (1) as an equivalent optimization problem exhibiting structure amenable to distributed implementation, which will allow us to split the original problem into simpler subtasks that can be executed in parallel while still guaranteeing convergence to the global MLE.

Since summands in (1) are coupled through $\mathbf{s}$, it is not straightforward to decompose the *unconstrained* optimization problem in (1). This prompts us to define the auxiliary variable $\mathbf{s}_j$ to represent the local estimate of $\mathbf{s}$ at sensor j, and consider the *constrained* optimization problem

$$\{\hat{\mathbf{s}}_j\}_{j=1}^J := \arg\min_{\mathbf{s}_j} -\sum_{j=1}^J \ln[p_j(\mathbf{x}_j; \mathbf{s}_j)]$$
$$\text{s. to } \mathbf{s}_j = \bar{\mathbf{s}}_b, \quad b \in \mathcal{B}, \quad j \in \mathcal{N}_b \tag{4}$$

where $\mathcal{B} \subseteq [1, J]$ is a subset of "bridge" sensors maintaining local vectors $\bar{\mathbf{s}}_b$ that are utilized to impose consensus among local variables $\mathbf{s}_j$ across all sensors. If, e.g., $\mathcal{B} \equiv [1, J]$, then (a1) and the constraint $\mathbf{s}_j = \bar{\mathbf{s}}_b, b \in \mathcal{B}, j \in \mathcal{N}_b$ will render $\mathbf{s}_j = \mathbf{s}_i \; \forall \, i, j$. In such a case (1) and (4) are equivalent in the sense that $\hat{\mathbf{s}}_j = \hat{\mathbf{s}}_{ML} \; \forall \, j \in [1, J]$. In fact, a milder requirement on $\mathcal{B}$ is sufficient to ensure equivalence of (1) and (4), as described in the following definition.

Definition 1: Set $\mathcal{B}$ is a subset of bridge sensors if and only if
(a) $\forall \, j \in [1, J]$ there exists at least one $b \in \mathcal{B}$ so that $b \in \mathcal{N}_j$; and
(b) If j_1 and j_2 are single-hop neighboring sensors, there must exist a bridge sensor b so that $b \in \mathcal{N}_{j1} \cap \mathcal{N}_{j2}$.

For the WSN in Fig. 1 (left) a possible selection of sensors forming a bridge sensor subset $\mathcal{B}$, obeying (a) and (b), is represented by the black nodes. For future reference, the set of bridge neighbors of the jth sensor will be denoted as $\mathcal{B}_j := \mathcal{N}_j \cap \mathcal{B}$, and its cardinality by $|\mathcal{B}_j|$ for $j = 1, \ldots, J$.

In words, condition (a) in Definition 1 ensures that every node has a bridge-sensor neighbor; while condition (b) ensures that all the bridge variables $\{\bar{\mathbf{s}}_b\}_{b \in \mathcal{B}}$ can reach consensus (become equal). Together, they provide a necessary and sufficient condition for the equivalence between (1) and (4) as asserted by the following proposition.

Proposition 1: The optimal solutions of (1) and (4) coincide; i.e.

$$\hat{\mathbf{s}}_{ML} = \hat{\mathbf{s}}_j, \quad \forall \, j \in [1, J]. \tag{5}$$

if and only if $\mathcal{B}$ is a subset of bridge sensors as in Definition 1.

Proof: See Appendix A.

Proposition 1 asserts that consensus can be achieved across all J sensors if and only if consensus is reached only among a subset of them. As will become apparent in Section V-A, this reduced-size subset $\mathcal{B}$ lowers the communication cost. Further, bridge sensors trade-off communication cost for robustness to sensor failures; i.e., increasing the number of bridge sensors improves robustness to sensor failures but also increases communication cost and vice versa. Interestingly, the problem in (4) exhibits a distributable structure, as we show in Section IV.

A. The Alternating-Direction Method of Multipliers

Here we show how to solve (1) by combining the method of multipliers with a block coordinate descent iteration [2, pp. 253-261]. This procedure will yield a distributed estimation algorithm whereby local iterates $\mathbf{s}_j(k)$ converge to the MLE $\hat{\mathbf{s}}_{ML}$.

The method of multipliers exploits the decomposable structure of the augmented Lagrangian. Let $\mathbf{v}_j^b$ denote the Lagrange multiplier associated with the constraint $\mathbf{s}_j = \bar{\mathbf{s}}_b$. The multipliers $\{\mathbf{v}_j^b\}_{b \in \mathcal{B}_j}$ are kept at the jth sensor. The augmented Lagrangian for (4) is given by

$$\mathcal{L}_a[\boldsymbol{s}, \bar{\mathbf{s}}, \mathbf{v}] = -\sum_{j=1}^J \ln[p_j(\mathbf{x}_j; \mathbf{s}_j)] + \sum_{b \in \mathcal{B}} \sum_{j \in \mathcal{N}_b} \left(\mathbf{v}_j^b\right)^T (\mathbf{s}_j - \bar{\mathbf{s}}_b)$$
$$+ \sum_{b \in \mathcal{B}} \sum_{j \in \mathcal{N}_b} \frac{c_j}{2} \|\mathbf{s}_j - \bar{\mathbf{s}}_b\|_2^2 \tag{6}$$

where $\boldsymbol{s} := \{\mathbf{s}_j\}_{j=1}^J$, $\bar{\mathbf{s}} := \{\bar{\mathbf{s}}_b\}_{b \in \mathcal{B}}$ and $\mathbf{v} := \left\{\mathbf{v}_j^b\right\}_{j \in [1,J]}^{b \in \mathcal{B}_j}$. The constants $\{c_j > 0\}_{j=1}^J$ are penalty coefficients corresponding to the constraints $\mathbf{s}_j = \bar{\mathbf{s}}_b, \forall \, b \in \mathcal{B}_j$. Recall that the jth sensor maintains the local estimate $\mathbf{s}_j$; if this sensor also belongs to the subset of bridge sensors, i.e., if $j \in \mathcal{B}$, it also maintains the consensus variable $\bar{\mathbf{s}}_b$. Combining the method of multipliers with a block coordinate descent iteration, we obtain the following result.

Proposition 2: For a time index k consider iterates $\mathbf{v}_j^b(k)$, $\mathbf{s}_j(k)$ and $\bar{\mathbf{s}}_b(k)$ defined by the recursions

$$\mathbf{v}_j^b(k) = \mathbf{v}_j^b(k-1) + c_j \left[\mathbf{s}_j(k) - \bar{\mathbf{s}}_b(k)\right], \quad b \in \mathcal{B}_j \tag{7}$$

$$\mathbf{s}_j(k+1) = \arg\min_{\mathbf{s}_j} \Bigg[-\ln p_j(\mathbf{x}_j; \mathbf{s}_j) + \sum_{b \in \mathcal{B}_j} \left[\mathbf{v}_j^b(k)\right]^T \left[\mathbf{s}_j - \bar{\mathbf{s}}_b(k)\right] + \sum_{b \in \mathcal{B}_j} \frac{c_j}{2} \left\|\mathbf{s}_j - \bar{\mathbf{s}}_b(k)\right\|_2^2 \Bigg] \tag{8}$$

$$\bar{\mathbf{s}}_b(k+1) = \sum_{j \in \mathcal{N}_b} \frac{1}{\sum_{\beta \in \mathcal{N}_b} c_\beta} \left[\mathbf{v}_j^b(k) + c_j \mathbf{s}_j(k+1)\right], \quad b \in \mathcal{B} \tag{9}$$

for all sensors $j \in [1, J]$; and let the initial values of the Lagrange multipliers $\left\{\mathbf{v}_j^b(-1)\right\}_{b \in \mathcal{B}_j}$, the local estimates $\{\mathbf{s}_j(0)\}_{j=1}^J$ and the consensus variables $\{\bar{\mathbf{s}}_b(0)\}_{b \in \mathcal{B}}$ be arbitrary. Assuming ideal communication links and the validity of (a1) and (a2), the iterates $\mathbf{s}_j(k)$ converge to the MLE as $k \to \infty$; i.e.

$$\lim_{k \to \infty} \mathbf{s}_j(k) = \lim_{k \to \infty} \bar{\mathbf{s}}_b(k) = \hat{\mathbf{s}}_{ML}, \quad \forall \, j \in [1, J], \quad b \in \mathcal{B}. \tag{10}$$

We then say that as $k \to \infty$ the WSN reaches consensus.

Proof: See Appendix B.

The recursions in (7)–(9) constitute our distributed (D-) MLE algorithm. All sensors $j \in [1, J]$ keep track of the local estimate $\mathbf{s}_j(k)$ along with the Lagrange multipliers $\{\mathbf{v}_j^b(k)\}_{b\in\mathcal{B}_j}$. The bridge sensors belonging to $\mathcal{B}$ also update the consensus enforcing variables $\bar{\mathbf{s}}_b(k)$. With reference to Algorithm 1, during the kth iteration, sensor j receives the consensus variables $\bar{\mathbf{s}}_b(k)$ from all its neighboring bridge sensors $b \in \mathcal{B}_j$. Based on these consensus variables, it updates the Lagrange multipliers $\{\mathbf{v}_j^b(k)\}_{b\in\mathcal{B}_j}$ using (7), which are then used to compute $\mathbf{s}_j(k+1)$ via (8). After determining $\mathbf{s}_j(k+1)$, sensor j transmits to each of its neighbors $b \in \mathcal{B}_j$ the vector $\mathbf{v}_j^b(k) + c_j\mathbf{s}_j(k+1)$; see also Fig. 1 (right). Each sensor $b \in \mathcal{B}$ receives the vectors $\mathbf{v}_j^b(k) + c_j\mathbf{s}_j(k+1)$ from all its neighbors $j \in \mathcal{N}_b$, and proceeds to compute $\bar{\mathbf{s}}_b(k+1)$ using (9). This completes the kth iteration, after which all sensors in $\mathcal{B}$ transmit $\bar{\mathbf{s}}_b(k+1)$ to all their neighbors $j \in \mathcal{N}_b$, which can then initialize the $(k+1)$-st iteration.

Note that the minimization required in (8) is unconstrained, and the corresponding cost function is strictly convex as per (a2) and the strict convexity of the Euclidean norm. Thus, the optimal solution $\mathbf{s}_j(k+1)$ of (8) is unique and can be obtained by finding the (unique) root of the cost function's gradient. Upon defining $\mathbf{f}_j(\mathbf{s}_j(k+1)) := \mathbf{s}_j(k+1) - 1/c_j|\mathcal{B}_j|[p_j(\mathbf{x}_j;\mathbf{s}_j(k+1))]^{-1}\nabla_{\mathbf{s}_j}p_j(\mathbf{x}_j;\mathbf{s}_j(k+1))$, this means that $\mathbf{s}_j(k+1)$ can be found as the unique solution of

$$\mathbf{f}_j(\mathbf{s}_j(k+1)) = -\frac{1}{c_j|\mathcal{B}_j|}\sum_{b\in\mathcal{B}_j}\mathbf{v}_b^j(k) + \frac{1}{|\mathcal{B}_j|}\sum_{b\in\mathcal{B}_j}\bar{\mathbf{s}}_b(k). \quad (11)$$

Equation (11) can be solved numerically at the jth sensor using, e.g., Newton's method.

Remark 2: The decentralized algorithms constructed in [1], [13], [18], [20] require knowing the desired estimator in closed form expressed in terms of averages. The recursions (7)–(9) and the resultant D-MLE in Algorithm 1 do not require a closed-form expression for the desired estimator but only the mild log-concavity assumption (a2). A general sum of *strictly* convex (and thus locally identifiable) functions was formulated in [11] and dual decomposition techniques were invoked to establish convergence and resilience of distributed iterative estimation to erasure links in the context of consensus averaging, i.e., for the sample mean estimation. The differences between the present formulation and [11] are: i) only global identifiability is required here; ii) bridge sensors offer flexibility to trade-off communication cost for tolerance to sensor failures ([11] can be seen as a special case where each sensor is a bridge sensor); and iii) since the approach in [11] can be viewed as a consensus averaging scheme with a proper weight matrix, it inherits its limitations in convergence speed and sensitivity to additive noise; see also discussion in Section V and Remark 4.

Algorithm 1: D-MLE

Initialize $\{\mathbf{s}_j(0)\}_{j=1}^J$, $\{\bar{\mathbf{s}}_b(0)\}_{b\in\mathcal{B}}$ and $\{\mathbf{v}_j^b(-1)\}_{j\in[1,J]}^{b\in\mathcal{B}_j}$ randomly.

for $k = 0, 1, \ldots$ **do**

Bridge sensors $b \in \mathcal{B}$: transmit $\bar{\mathbf{s}}_b(k)$ to neighbors in $\mathcal{N}_b$

All $j \in [1, J]$: update $\{\mathbf{v}_j^b(k)\}_{b\in\mathcal{B}_j}$ using **(7)**.

All $j \in [1, J]$: update $\mathbf{s}_j(k+1)$ using **(8)**.

All $j \in [1, J]$: transmit $\mathbf{v}_j^b(k) + c_j\mathbf{s}_j(k+1)$ to each $b \in \mathcal{B}_j$

Bridge sensors $b \in \mathcal{B}$: compute $\bar{\mathbf{s}}_b(k+1)$ through **(9)**.

end for

B. Communication Errors

When the communication links are corrupted by additive noise as in (3), the neighboring variables used in (7)–(9) have to be modified accordingly. The variable $\mathbf{s}_j(k)$ in (7) for instance is local; but the term $\bar{\mathbf{s}}_b(k)$ is received from the bth bridge neighbor and has to be replaced by $\bar{\mathbf{s}}_b(k) + \boldsymbol{\eta}_j^b(k)$ [cf. (3)]. Altogether, (7)–(9) should be replaced by (12)–(14), shown at the bottom of the next page. Since $\mathbf{s}_j(k+1)$ and $\bar{\mathbf{s}}_b(k+1)$ in (12)–(14) are obtained as the optimal solution of pertinent minimization problems [cf. Appendix B], the D-MLE algorithm exhibits noise resilience. In the presence of noise (12)–(14) can be thought of as comprising a stochastic gradient algorithm; e.g., [2, Sec.7.8]. This suggests that noise causes $\mathbf{s}_j(k)$ to fluctuate around the optimal solution $\hat{\mathbf{s}}_{ML}$ with the magnitude of fluctuations being proportional to the noise variance. However, $\mathbf{s}_j(k)$ is guaranteed to remain within a ball around $\hat{\mathbf{s}}_{ML}$ with high probability. This should be contrasted with [19] which suffers from catastrophic noise propagation. Resilience to noise in the communication links will become apparent in Sections III-C and III-D where we apply the D-MLE algorithm to interesting estimation setups.

C. Example 1—Linear Gaussian Model

Algorithm 1 is applicable to the linear Gaussian model where $\mathbf{s}$ is to be estimated from observations $\{\mathbf{x}_j = \mathbf{H}_j\mathbf{s}+\mathbf{n}_j\}_{j=1}^J$ and $\mathbf{n}_j$ is AGN with covariance matrix $\boldsymbol{\Sigma}_{n_jn_j}$. The pdf of $\mathbf{x}_j$ is thus

$$p_j(\mathbf{x}_j;\mathbf{s}) = \frac{1}{(2\pi)^{L_j/2}\det^{1/2}(\boldsymbol{\Sigma}_{n_jn_j})} \times \exp\left[-\frac{1}{2}(\mathbf{x}_j - \mathbf{H}_j\mathbf{s})^T\boldsymbol{\Sigma}_{n_jn_j}^{-1}(\mathbf{x}_j - \mathbf{H}_j\mathbf{s})\right]. \quad (15)$$

The log-concavity assumption (a2) is satisfied by $p_j(\mathbf{x}_j;\mathbf{s})$ since $\ln[p_j(\mathbf{x}_j;\mathbf{s})] = -1/2(\mathbf{x}_j - \mathbf{H}_j\mathbf{s})^T\boldsymbol{\Sigma}_{n_jn_j}^{-1}(\mathbf{x}_j - \mathbf{H}_j\mathbf{s})$ is a quadratic form with $\boldsymbol{\Sigma}_{n_jn_j}$ positive semidefinite. In this case, the minimization in (8) can be solved in closed-form:

$$\mathbf{s}_j(k+1) = \left[\mathbf{H}_j^T\boldsymbol{\Sigma}_{n_jn_j}^{-1}\mathbf{H}_j + c_j|\mathcal{B}_j|\mathbf{I}_p\right]^{-1} \times \left[\mathbf{H}_j^T\boldsymbol{\Sigma}_{n_jn_j}^{-1}\mathbf{x}_j - \sum_{b\in\mathcal{B}_j}\mathbf{v}_j^b(k) + c_j\sum_{b\in\mathcal{B}_j}\bar{\mathbf{s}}_b(k)\right]. \quad (16)$$

The matrix inversion in (16) can be performed off-line at each sensor since all quantities involved are known. This emphasizes the simplicity of Algorithm 1 especially for linear models.

D. Example 2–Quantized Observations

An interesting twist on the previous example is when due to limited sensing capabilities the sensors produce a coarsely quantized version of $\mathbf{x}_j$[12], [17]. Specifically, consider a K_j-element convex tessellation of $\mathbb{R}^{L_j}$ where the sets $\{\mathcal{Q}_{jk}\}_{k=1}^{K_j}$ are convex. Vector quantization of $\mathbf{x}_j$ produces the observation $\mathbf{b}_{jk} = \mathbf{1}\{\mathbf{x}_j \in \mathcal{Q}_{jk}\}$ where $\mathbf{1}$ denotes the indicator function; i.e., vector $\mathbf{b}_{jk}$ has binary $0/1$ entries (1 if and only if $\mathbf{x}_j$ falls in the quantization region $\mathcal{Q}_{jk}$). The probability mass function of $\mathbf{b}_{jk}$ parameterized by the unknown vector $\mathbf{s}$ is

$$\Pr(\mathbf{b}_{jk};\mathbf{s}) = \int_{\mathbf{x}_j\in\mathcal{Q}_{jk}} p_j(\mathbf{x}_j;\mathbf{s})d\mathbf{x}_j. \quad (17)$$

with $p_j(\mathbf{x}_j;\mathbf{s})$ as in (15). The integral of the log-concave $p_j(\mathbf{x}_j;\mathbf{s})$ over the convex set $\mathcal{Q}_{jk}$ is also log-concave establishing that (a2) is valid in this case too [12]. The MLE can be found by applying D-MLE to minimize in a distributed fashion the cost in (1), where $p(\mathbf{x}_j;\mathbf{s})$ is substituted by

$\Pr(\mathbf{b}_j;\mathbf{s}) = \sum_{l=1}^{K_j} \Pr(\mathbf{b}_{jl};\mathbf{s})\delta(\mathbf{b}_j - \mathbf{b}_{jl})$, where $\delta(\cdot)$ is the Kronecker delta. The resulting local minimization problem in (8) takes the form

$$\mathbf{s}_j(k+1) = \arg\min_{\mathbf{s}_j} \left[-\ln \Pr(\mathbf{b}_j;\mathbf{s}_j) + \sum_{b\in\mathcal{B}_j} \left(\mathbf{v}_j^b(k)\right)^T \times (\mathbf{s}_j - \bar{\mathbf{s}}_b(k)) + \sum_{b\in\mathcal{B}_j} \frac{c_j}{2}\|\mathbf{s}_j - \bar{\mathbf{s}}_b(k)\|_2^2 \right]. \quad (18)$$

Unlike (16) the local estimates $\mathbf{s}_j(k+1)$ are not computable in closed form.

For a WSN with $J = 60$ sensors, we apply the D-MLE in Algorithm 1 to the problem introduced in this section. Nodes in the WSN are randomly placed in the unit square $[0,1]\times[0,1]$ with uniform distribution. Each sensor collects $L_j = 5$ observations, while $\mathbf{s}$ consists of $p = 2$ parameters. The entries of $\mathbf{H}_j$ are random uniformly distributed over $[-0.5, 0.5]$ and vectors $\{\mathbf{n}_j\}_{j=1}^J$ are zero-mean AGN with $\mathbf{\Sigma}_{n_j n_j} = 0.5\mathbf{I}_{p\times p}$. The quantizer at the jth sensor splits $\mathbb{R}^5$ into $K_j = 32$ regions defined as $\mathcal{Q}_{j1} = \{\mathbf{x}_{j,1} > 0\}\cup\cdots\cup\{\mathbf{x}_{j,5} > 0\};\ldots;\mathcal{Q}_{j32} = \{\mathbf{x}_{j,1} < 0\}\cup\cdots\cup\{\mathbf{x}_{j,5} < 0\}\ \forall\, j = 1,\ldots,J$. The penalty coefficients are set to $c_j = 10/|\mathcal{B}_j|$. The performance metric considered is the normalized error defined as

$$e_{\text{norm}}(k) := \sum_{j=1}^{J} \frac{\|\mathbf{s}_j(k) - \hat{\mathbf{s}}_{ML}\|_2}{\|\hat{\mathbf{s}}_{ML}\|_2}. \quad (19)$$

When the communication links are ideal, Fig. 2 (top)-(bottom) depicts $e_{\text{norm}}(k) \to 0$ as $k \to \infty$, corroborating the result in Proposition 2. We then add either reception or quantization noise and average $e_{\text{norm}}(k)$ over 50 independent realizations of the D-MLE. For analog communications in noise as in (n1), we set for simplicity $\mathbf{\Sigma}_{\eta_j\eta_j} = \sigma^2\mathbf{I}_p$, and adjust the noise variance σ^2 so that $\text{SNR} := 10\log_{10}\hat{\mathbf{s}}_{ML}/p/\sigma^2 = 20$ dB and $\text{SNR} = 15$ dB. Interestingly, in the presence of reception noise the average $e_{\text{norm}}(k)$ exhibits an error floor for the D-MLE [see Fig. 2 (top)]. This indicates that errors do not explode as in the consensus average approach [19], but instead converge to a bounded value. Error floor is also observed in Fig. 2 (bottom) where the sensors quantize their local estimates and/or consensus variables before digital transmission to the corresponding neighbors. Fig. 2 (bottom) depicts the error curves for $m_j = m = 5, 10, \infty$ quantization bits with $j \in [1, 60]$.

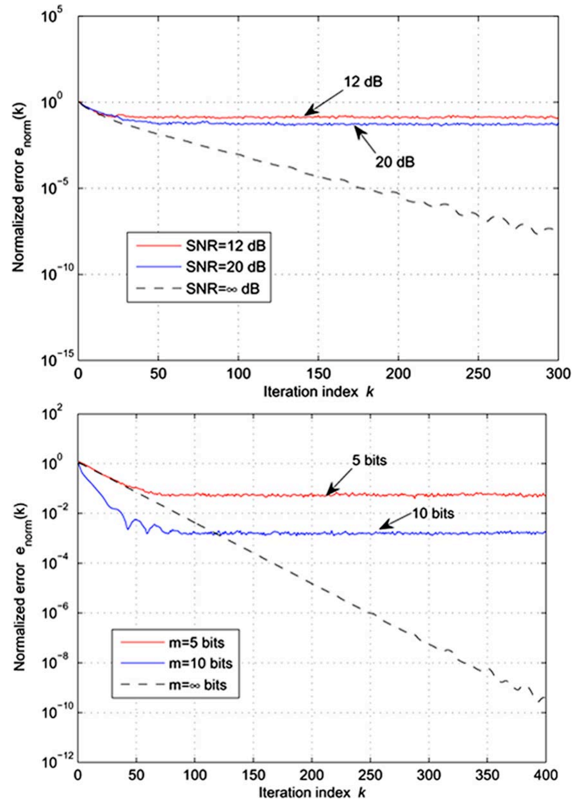

Fig. 2. Normalized error $e_{\text{norm}}(k)$ versus k for D-MLE in presence of (top) reception noise with $\text{SNR} = 12, 20$ and ∞ dB, and (bottom) quantization noise using $m = 5, 10$ and ∞ number of bits.

IV. Distributed BLUE

In this section, we consider decentralized estimation of $\hat{\mathbf{s}}_{BL}$ in (s2), under (a1). As in Section III we write $\hat{\mathbf{s}}_{BL}$ as the solution of a constrained convex minimization problem and utilize the alternating direction method of multipliers to obtain a decentralized algorithm with iterates $\mathbf{s}_j(k)$ converging to $\hat{\mathbf{s}}_{BL}$. This algorithm is subsequently used to derive a distributed linear estimator that is amenable to convergence analysis in the presence of noise.

A. The D-BLUE Algorithm

For the linear Gaussian model, we have seen how to deduce the D-BLUE from the D-MLE and proceed as in Section III; however, we will find it useful for general data models [cf. (s2)] to derive D-BLUE by viewing $\hat{\mathbf{s}}_{BL}$ in (2) as the minimizer of a different quadratic function detailed in the next lemma.

Lemma 1: The BLUE in (2) can be written as

$$\hat{\mathbf{s}}_{BL} = \arg\min_{\mathbf{s}\in\mathbb{R}^{p\times 1}} \sum_{j=1}^{J} \left\| \mathbf{\Sigma}_{x_jx_j}^{-1/2}\mathbf{H}_j\mathbf{s} - \mathbf{\Sigma}_{x_jx_j}^{-1/2}\mathbf{x}_j \right\|_2^2. \quad (20)$$

Proof: See Appendix C.

Similar to (1), the minimization in (20) cannot be implemented in a distributed fashion motivating the introduction of local estimates $\mathbf{s}_j$ and consensus enforcing variables $\bar{\mathbf{s}}_b$ to reformulate (20) as shown in the following lemma.

$$\mathbf{v}_j^b(k) = \mathbf{v}_j^b(k-1) + c_j\left[\mathbf{s}_j(k) - \left(\bar{\mathbf{s}}_b(k) + \boldsymbol{\eta}_j^b(k)\right)\right], \quad b\in\mathcal{B}_j \quad (12)$$

$$\mathbf{s}_j(k+1) = \arg\min_{\mathbf{s}_j}\left[-\ln p_j(\mathbf{x}_j;\mathbf{s}_j) + \sum_{b\in\mathcal{B}_j}\left[\left(\mathbf{v}_j^b(k)\right)^T\mathbf{s}_j - \left(\bar{\mathbf{s}}_b(k) + \boldsymbol{\eta}_j^b(k)\right)\right] + \sum_{b\in\mathcal{B}_j}\frac{c_j}{2}\left\|\mathbf{s}_j - \left(\bar{\mathbf{s}}_b(k) + \boldsymbol{\eta}_j^b(k)\right)\right\|_2^2\right] \quad (13)$$

$$\bar{\mathbf{s}}_b(k+1) = \sum_{j\in\mathcal{N}_b}\frac{1}{\sum_{\beta\in\mathcal{N}_b} c_\beta}\left(\mathbf{v}_j^b(k) + c_j\left(\mathbf{s}_j(k+1) + \boldsymbol{\eta}_b^j(k+1)\right)\right), \quad b\in\mathcal{B}. \quad (14)$$

Lemma 2: For a set $\mathcal{B}$ of bridge sensors as in Definition 1, the minimization in (20) is equivalent to

$$\{\hat{\mathbf{s}}_j\}_{j=1}^J := \arg\min_{\mathbf{s}_j} \sum_{j=1}^J \left\| \boldsymbol{\Sigma}_{x_jx_j}^{-1/2}\mathbf{H}_j\mathbf{s}_j - \boldsymbol{\Sigma}_{x_jx_j}^{-1/2}\mathbf{x}_j \right\|_2^2$$
$$\text{s. to } \mathbf{s}_j = \bar{\mathbf{s}}_b, \quad b \in \mathcal{B}, \quad j \in \mathcal{N}_b \tag{21}$$

in the sense that $\hat{\mathbf{s}}_{BL} = \hat{\mathbf{s}}_j \ \forall\, j \in [1, J]$.

The proof of Lemma 2 is similar to the proof of Proposition 1 and we omit it for brevity. As with D-MLE, the jth sensor in (21) maintains the local estimate $\mathbf{s}_j$ with the bth bridge sensor also maintaining $\bar{\mathbf{s}}_b$. The augmented Lagrangian can now be written as

$$\mathcal{L}_a(\boldsymbol{s}, \bar{\mathbf{s}}, \mathbf{v}) = \sum_{j=1}^J \left\| \boldsymbol{\Sigma}_{x_jx_j}^{-1/2}\mathbf{H}_j\mathbf{s}_j - \boldsymbol{\Sigma}_{x_jx_j}^{-1/2}\mathbf{x}_j \right\|_2^2 + \sum_{b\in\mathcal{B}}\sum_{j\in\mathcal{N}_b} \left(\mathbf{v}_j^b\right)^T (\mathbf{s}_j - \bar{\mathbf{s}}_b) + \sum_{b\in\mathcal{B}}\sum_{j\in\mathcal{N}_b} \frac{c_j}{2}\|\mathbf{s}_j - \bar{\mathbf{s}}_b\|_2^2. \tag{22}$$

Proceeding as in Section III-A, we will minimize (22) using the method of alternating direction multipliers [cf. Appendix B], and derive the D-BLUE algorithm summarized next.

Proposition 3: For all $j \in [1, J]$, consider iterates $\mathbf{v}_j^b(k)$, $\mathbf{s}_j(k)$ and $\bar{\mathbf{s}}_b(k)$ defined by the recursions

$$\mathbf{v}_j^b(k) = \mathbf{v}_j^b(k-1) + c_j[\mathbf{s}_j(k) - \bar{\mathbf{s}}_b(k)], \quad b \in \mathcal{B}_j \tag{23}$$
$$\mathbf{s}_j(k+1) = \hat{\mathbf{x}}_j - \mathbf{B}_j^{-1}\left(\sum_{b\in\mathcal{B}_j}\mathbf{v}_j^b(k) - c_j\sum_{b\in\mathcal{B}_j}\bar{\mathbf{s}}_b(k)\right) \tag{24}$$
$$\bar{\mathbf{s}}_b(k+1) = \sum_{j\in\mathcal{N}_b}\frac{1}{\sum_{\beta\in\mathcal{N}_b}c_\beta}\left(\mathbf{v}_j^b(k) + c_j\mathbf{s}_j(k+1)\right), \quad b \in \mathcal{B} \tag{25}$$

with $\mathbf{B}_j := 2\mathbf{H}_j^T\boldsymbol{\Sigma}_{x_jx_j}^{-1}\mathbf{H}_j + c_j|\mathcal{B}_j|\mathbf{I}_p$ and $\hat{\mathbf{x}}_j := \mathbf{B}_j^{-1}2\mathbf{H}_j^T\boldsymbol{\Sigma}_{x_jx_j}^{-1}\mathbf{x}_j$. Assuming ideal communication links and the validity of (a1) and (a2), the iterates $\mathbf{s}_j(k)$ converge to the BLUE as $k \to \infty$; i.e.

$$\lim_{k\to\infty}\mathbf{s}_j(k) = \lim_{k\to\infty}\bar{\mathbf{s}}_b(k) = \hat{\mathbf{s}}_{BL} \quad \forall\, j \in [1, J] \tag{26}$$

for arbitrary initial values $\{\mathbf{v}_j^b(-1)\}_{b\in\mathcal{B}_j}$, $\{\mathbf{s}_j(0)\}_{j=1}^J$ and $\{\bar{\mathbf{s}}_b(-1)\}_{b\in\mathcal{B}}$.

Proof: Follows easily by mimicking the steps used in the proof of Proposition 2.

Vector $\hat{\mathbf{x}}_j$ can be regarded as a regularized version of the local BLUE. Indeed, the jth sensor's local BLUE is $\hat{\mathbf{x}}_{j,BL} := \left(\mathbf{H}_j^T\boldsymbol{\Sigma}_{x_jx_j}^{-1}\mathbf{H}_j\right)^{-1}\mathbf{H}_j^T\boldsymbol{\Sigma}_{x_jx_j}^{-1}\mathbf{x}_j$, which except for the positive definite $c_j|\mathcal{B}_j|\mathbf{I}_p$ in the definition of $\mathbf{B}_j$ coincides with $\hat{\mathbf{x}}_j = \left(2\mathbf{H}_j^T\boldsymbol{\Sigma}_{x_jx_j}^{-1}\mathbf{H}_j + c_j|\mathcal{B}_j|\mathbf{I}_p\right)^{-1}2\mathbf{H}_j^T\boldsymbol{\Sigma}_{x_jx_j}^{-1}\mathbf{x}_j$. The regularization term is needed because we require full column rank for $\mathbf{H}$ (to ensure global identifiability) but not for each individual $\mathbf{H}_j$.

Recursions (23)–(25) are used to implement D-BLUE as summarized in Algorithm 2. Since each sensor $j \in [1, J]$ has available the vector $\mathbf{H}_j^T\boldsymbol{\Sigma}_{x_jx_j}^{-1}\mathbf{x}_j$ along with the Lagrange multipliers $\{\mathbf{v}_j^b(k-1)\}_{b\in\mathcal{B}_j}$, and receives the consensus variables $\bar{\mathbf{s}}_b(k)$ from all its bridge neighbors $b \in \mathcal{B}_j$, it is able to update $\{\mathbf{v}_j^b(k)\}_{b\in\mathcal{B}_j}$ through (23) and compute $\mathbf{s}_j(k+1)$ using (24). Afterwards, sensor j transmits to all its bridge-neighbors the vectors $\mathbf{v}_j^b(k) + c_j\mathbf{s}_j(k+1)$ with $b \in \mathcal{B}_j$. To complete the kth iteration, every sensor $b \in \mathcal{B}$ receives the vectors $\{\mathbf{v}_j^b(k) + c_j\mathbf{s}_j(k+1)\}_{j\in\mathcal{N}_b}$ to form $\bar{\mathbf{s}}_b(k+1)$ using (25).

Remark 3: Since matrix $\mathbf{B}_j$ is time invariant, the jth sensor can perform the inversion off-line, e.g., during the WSN start-up phase. The computational cost for computing $\mathbf{s}_j(k+1)$ thus amounts to $O(p^2)$ for the matrix-vector multiplication. To evaluate the communication cost of Algorithm 2 notice first that during the kth iteration every sensor $j \in [1, J]$ sends to all its bridge-neighbors the vectors $\mathbf{v}_j^b(k) + c_j\mathbf{s}_j(k+1) \in \mathbb{R}^{p\times 1}$ with $b \in \mathcal{B}_j\backslash\{j\}$, where $\mathcal{S}1\backslash\mathcal{S}2$ denotes set subtraction of $\mathcal{S}2$ from $\mathcal{S}1$. The sensors belonging to the bridge-sensor set $\mathcal{B}$ also transmit their consensus variable $\bar{\mathbf{s}}_b(k)$ to all their neighbors. Note that for a sensor $j \in [1, J]\backslash\mathcal{B}$ it holds that $|\mathcal{B}_j\backslash\{j\}| = |\mathcal{B}_j|$; but for a sensor $b \in \mathcal{B}$ it holds that $|\mathcal{B}_b\backslash\{b\}| = |\mathcal{B}_b| - 1$. Thinking along these lines we deduce that each sensor has to transmit $p|\mathcal{B}_j|$ scalars per iteration. Thus, the amount of information each sensor has to communicate per iteration is $O(p)$, which is intuitively reasonable since each sensor wishes to form the $p \times 1$ vector $\hat{\mathbf{s}}_{BL}$.

Algorithm 2: D-BLUE

Initialize randomly $\{\mathbf{s}_j(0)\}_{j=1}^J$, $\{\bar{\mathbf{s}}_b(0)\}_{b\in\mathcal{B}}$ and $\{\mathbf{v}_j^b(-1)\}_{j\in[1,J]}^{b\in\mathcal{B}_j}$.

Compute the matrices $\mathbf{B}_j^{-1}$.

for $k = 0, 1, \ldots$ **do**

Bridge sensors $b \in \mathcal{B}$: transmit $\bar{\mathbf{s}}_b(k)$ to neighbors in $\mathcal{N}_b$;

All $j \in [1, J]$: update $\{\mathbf{v}_j^b(k)\}_{b\in\mathcal{B}_j}$ using **(23)**;

All $j \in [1, J]$: update $\mathbf{s}_j(k+1)$ using **(24)**;

All $j \in [1, J]$: transmit $\mathbf{v}_j^b(k) + c_j\mathbf{s}_j(k+1)$ to each $b \in \mathcal{B}_j$;

Bridge sensors $b \in \mathcal{B}$: compute $\bar{\mathbf{s}}_b(k+1)$ via **(25)**.

end for

V. NOISE-ROBUST D-BLUE

It is worth stressing that the D-BLUE recursions in Proposition 3 are linear in the problem variables. In fact, updating these variables, e.g., $\{\mathbf{s}_j(k)\}_{j=1}^J$, resembles that of a vector autoregressive (AR) process. This viewpoint will prove helpful to analyze D-BLUE in the presence of noise and develop noise-resilient versions of it. To this end, we will find it useful to reformulate the D-BLUE recursions in (23)–(25).

A. Multiplier-Free D-BLUE

Here we eliminate the Lagrange multipliers $\mathbf{v}_j^b$ and use specific initialization to rewrite the recursions (23)–(25) more compactly as suggested in the following lemma.

Lemma 3: Initialize the recursions (23)–(25) specifically with $\left\{\mathbf{v}_b^j(-1) = \mathbf{0}\right\}_{b\in\mathcal{B}_j}^{j\in[1,J]}$, $\{\bar{\mathbf{s}}_b(-1) = \mathbf{0}\}_{b\in\mathcal{B}}$ and $\{\mathbf{s}_j(0) = \hat{\mathbf{x}}_j\}_{j=1}^J$, where $\hat{\mathbf{x}}_j$ is the locally regularized BLUE at the jth sensor. The local iterates $\mathbf{s}_j(k)$ and the consensus enforcing variables $\bar{\mathbf{s}}_b(k)$ in Proposition 3 can then be rewritten for $k \geq 0$ as

$$\mathbf{s}_j(k+1) = \left(\mathbf{I}_p - c_j|\mathcal{B}_j|\mathbf{B}_j^{-1}\right)\mathbf{s}_j(k) + c_j\mathbf{B}_j^{-1} \times \sum_{b\in\mathcal{B}_j}[2\bar{\mathbf{s}}_b(k) - \bar{\mathbf{s}}_b(k-1)], \quad j \in [1, J] \tag{27}$$
$$\bar{\mathbf{s}}_b(k+1) = \sum_{j\in\mathcal{N}_b}\frac{c_j}{\sum_{\beta\in\mathcal{N}_b}c_\beta}\mathbf{s}_j(k+1), \quad b \in \mathcal{B}. \tag{28}$$

Proof: See Appendix D.

Lemma 3 shows that with carefully chosen initial conditions, (23)–(25) reduce to the multiplier-free pair of equations (27)–(28). Relative to (25), the consensus variable $\bar{\mathbf{s}}_b(k+1)$ in (28) is a weighted average of the neighborhood estimates $\{\mathbf{s}_j(k+1)\}_{j\in\mathcal{N}_b}$. Furthermore, the original observations embedded in the local estimates $\{\hat{\mathbf{x}}_j\}_{j=1}^J$ appear in (27)–(28) only through the initial conditions.

In the alternative formulation of D-BLUE obtained from Lemma 3 the kth iteration starts with all sensors $j \in [1, J]$ receiving the vectors $2\bar{\mathbf{s}}_b(k)-\bar{\mathbf{s}}_b(k-1)$ from their bridge-neighbors to update the local estimate $\mathbf{s}_j(k+1)$ using (27). Then, the bridge-sensors $b \in \mathcal{B}$ receive from their neighbors the vectors $\{\mathbf{s}_j(k+1)\}_{j\in\mathcal{N}_b}$ to update $\bar{\mathbf{s}}_b(k+1)$ using (28), and finally form $2\bar{\mathbf{s}}_b(k+1) - \bar{\mathbf{s}}_b(k)$ that they disseminate for their neighbors to start the $(k+1)$-st iteration.

From Lemma 3 it can be seen that $\mathbf{s}_j(k)$ is a second-order vector AR process with specific initial conditions. Careful examination of (27) and (28) reveals that each sensor, say the jth, updates its local estimate $\mathbf{s}_j(k+1)$ using information from neighboring sensors within a radius of two hops. Indeed, $\mathbf{s}_j(k+1)$ is updated using the consensus variables $\bar{\mathbf{s}}_b(k)$ and $\bar{\mathbf{s}}_b(k-1)$ for $b \in \mathcal{B}_j$, which are formed using the local estimates of all sensors within the set $\{\mathcal{N}_b\}_{b\in\mathcal{B}_j}$. This set contains all the sensors within a distance of either a single hop or two hops from sensor j.

Remark 4: The local updates of existing approaches in [1], [8], [11], [13], [18], [20] either have a memory of a single time step, or utilize updating information which is received only from single-hop neighbors. D-BLUE on the other hand, has the potential of achieving improved convergence rates because it utilizes more information across time and across space. One might expect that the price paid for improved convergence is increased communication and/or computational cost; however, this is not the case. Recall that decentralized computation of BLUE in [20] for the special case of a linear-Gaussian model $\mathbf{x}_j = \mathbf{H}_j\mathbf{s} + \mathbf{n}_j$ requires two separate consensus averaging algorithms to determine the matrix $\sum_{j=1}^J \mathbf{H}_j^T\mathbf{\Sigma}_{n_jn_j}^{-1}\mathbf{H}_j$ and the vector $\sum_{j=1}^J \mathbf{H}_j^T\mathbf{\Sigma}_{n_jn_j}^{-1}\mathbf{x}_j$, with computational complexity $O(p^2)$. However, the communication cost per iteration in D-BLUE is reduced from $O(p^2)$ to $O(p)$. Indeed, [20] requires communication of $p\times p$ matrices, while sensors in D-BLUE communicate $p\times 1$ vectors. Communication cost of $O(p)$ is also exhibited by the decentralized scheme in [1].

B. Differences-Based D-BLUE in the Presence of Noise

Building on (27) and (28) we will see in this section how to derive a provably noise-resilient version of D-BLUE. Instrumental to this derivation will be the noisy counterpart of $\mathbf{s}_j(k)$ in (27), that we denote by $\boldsymbol{\phi}_j(k)$ and is maintained as usual at the jth sensor. We will prove that successive differences of $\boldsymbol{\phi}_j(k)$ converge to the BLUE; i.e., $\lim_{k\to\infty}[\boldsymbol{\phi}_j(k+1)-\boldsymbol{\phi}_j(k)] = \hat{\mathbf{s}}_{BL}$. Intuitively, noise terms that propagate from $\boldsymbol{\phi}_j(k)$ to $\boldsymbol{\phi}_j(k+1)$ cancel when considering the difference $\boldsymbol{\phi}_j(k+1)-\boldsymbol{\phi}_j(k)$, thus achieving the desired robustness to noise. This is akin to the noise suppression effected also in the local updates of coupled oscillators in [1], where a *continuous-time* differential (state) equation is involved per sensor, and the information is encoded in the derivative of the state. The desired *discrete-time* recursion for $\boldsymbol{\phi}_j(k)$ and its relationship with (27) and (28) is introduced in the following lemma.

Lemma 4: If $\boldsymbol{\phi}_j(0) = \hat{\mathbf{x}}_j$ and $\boldsymbol{\phi}_j(-1) = \boldsymbol{\phi}_j(-2) = \mathbf{0}$, the second-order recursions

$$\boldsymbol{\phi}_j(k+1) = \hat{\mathbf{x}}_j + \left(\mathbf{I} - c_j|\mathcal{B}_j|\mathbf{B}_j^{-1}\right)\boldsymbol{\phi}_j(k) + c_j\mathbf{B}_j^{-1} \times \sum_{b\in\mathcal{B}_j}\left[2\bar{\boldsymbol{\phi}}_b(k) - \bar{\boldsymbol{\phi}}_b(k-1)\right] \tag{29}$$

$$\bar{\boldsymbol{\phi}}_b(k+1) = \sum_{j\in\mathcal{N}_b}\frac{c_j}{\sum_{\beta\in\mathcal{N}_b} c_\beta}\boldsymbol{\phi}_j(k+1), \quad b\in\mathcal{B}, \quad k\geq -1 \tag{30}$$

yield iterates $\boldsymbol{\phi}_j(k)$ and $\bar{\boldsymbol{\phi}}_b(k)$ whose differences $\delta\boldsymbol{\phi}_j(k) := \boldsymbol{\phi}_j(k) - \boldsymbol{\phi}_j(k-1)$ and $\delta\bar{\boldsymbol{\phi}}_b(k) := \bar{\boldsymbol{\phi}}_b(k) - \bar{\boldsymbol{\phi}}_b(k-1)$ equal the iterates $\mathbf{s}_j(k)$ and $\bar{\mathbf{s}}_b(k)$ produced by (27) and (28), respectively.

Proof: See Appendix E.

Upon recalling from Proposition 3 and Lemma 3 that $\lim_{k\to\infty}\mathbf{s}_j(k) = \hat{\mathbf{s}}_{BL}$, we obtain readily that $\lim_{k\to\infty}[\boldsymbol{\phi}_j(k) - \boldsymbol{\phi}_j(k-1)] = \lim_{k\to\infty}\delta\boldsymbol{\phi}_j(k) = \lim_{k\to\infty}\delta\bar{\boldsymbol{\phi}}_b(k) = \hat{\mathbf{s}}_{BL}$ when the communication links are ideal; i.e., successive differences of the state in (29) converge to the BLUE.

Recursions (29) and (30) are similar in form to (27) and (28), and can be implemented in a decentralized fashion as described in Section V-A. Furthermore, upon defining the quantities $\bar{\boldsymbol{\psi}}_b(k) := 2\bar{\boldsymbol{\phi}}_b(k) - \bar{\boldsymbol{\phi}}_b(k-1)$ and $\boldsymbol{\psi}_j(k+1) := 2\boldsymbol{\phi}_j(k+1) - \boldsymbol{\phi}_j(k)$, we can rewrite (29) and (30) as

$$\boldsymbol{\phi}_j(k+1) = \hat{\mathbf{x}}_j + \left(\mathbf{I} - c_j|\mathcal{B}_j|\mathbf{B}_j^{-1}\right)\boldsymbol{\phi}_j(k) + c_j\mathbf{B}_j^{-1} \times \sum_{b\in\mathcal{B}_j}\bar{\boldsymbol{\psi}}_b(k) \tag{31}$$

$$\bar{\boldsymbol{\psi}}_b(k+1) = \sum_{j\in\mathcal{N}_b}\frac{c_j}{\sum_{\beta\in\mathcal{N}_b} c_\beta}\boldsymbol{\psi}_j(k+1), \quad b\in\mathcal{B}. \tag{32}$$

Beyond ideal links, (31) and (32) will enable robustness in the presence of reception or quantization noise. In the noisy case, distributed implementation of (31) and (32) involves two steps: (i) all sensors $j\in[1,J]$ receive the vectors $\bar{\boldsymbol{\psi}}_b(k) + \boldsymbol{\eta}_j^b(k)$ from $b\in\mathcal{B}_j$ to form a (noisy) iterate $\boldsymbol{\phi}_j(k+1)$; and (ii) bridge sensors receive the vector $\boldsymbol{\psi}_j(k+1) + \bar{\boldsymbol{\eta}}_b^j(k+1)$ from $j\in\mathcal{N}_b$ to form the (noisy) iterate $\bar{\boldsymbol{\psi}}_b(k+1)$. Explicitly written, (31) and (32) in noise are replaced by

$$\boldsymbol{\phi}_j(k+1) = \hat{\mathbf{x}}_j + \left(\mathbf{I} - c_j|\mathcal{B}_j|\mathbf{B}_j^{-1}\right)\boldsymbol{\phi}_j(k) + c_j\mathbf{B}_j^{-1} \times \sum_{b\in\mathcal{B}_j}\bar{\boldsymbol{\psi}}_b(k) + c_j\mathbf{B}_j^{-1}\sum_{b\in\mathcal{B}_j\setminus\{j\}}\boldsymbol{\eta}_j^b(k) \tag{33}$$

$$\bar{\boldsymbol{\psi}}_b(k+1) = \sum_{j\in\mathcal{N}_b}\frac{c_j}{\sum_{\beta\in\mathcal{N}_b} c_\beta}\left[\boldsymbol{\psi}_j(k+1)\right] + \sum_{j\in\mathcal{N}_b, j\neq b}\frac{c_j}{\sum_{\beta\in\mathcal{N}_b} c_\beta}\bar{\boldsymbol{\eta}}_b^j(k+1), \quad b\in\mathcal{B}. \tag{34}$$

Notice that if the jth sensor is a bridge sensor, it contains a noise-free version of $\bar{\boldsymbol{\psi}}_j(k)$; that is why we excluded j from the second sum in (33). Similarly, the bth bridge sensor has a noiseless version of $\boldsymbol{\psi}_b(k+1)$ and for this reason we excluded b from the second sum in (34). Recursions (33) and (34) constitute our robust (R) D-BLUE which we tabulate as Algorithm 3. From (33) and (34) it can be seen that by having the jth sensor transmitting $\boldsymbol{\psi}_j(k)$ and the bth sensor $\bar{\boldsymbol{\psi}}_b(k)$, instead of transmitting $\boldsymbol{\phi}_j(k)$ and $\bar{\boldsymbol{\phi}}_b(k)$ individually as (29) and (30) would suggest, the noise present in the updating process is reduced. In what follows we quantify this noise resilience based on the global recursion formed by concatenating (33) for $j=1,\dots,J$.

To this end, let us define the matrices $\mathbf{A}_1 := (\mathrm{diag}(c_1|\mathcal{B}_1|\dots c_J|\mathcal{B}_J|)\otimes \mathbf{I}_p)\mathbf{B}^{-1} - 2\mathbf{B}^{-1}\mathbf{W}_E$ and $\mathbf{A}_2 := \mathbf{B}^{-1}\mathbf{W}_E$, with $\mathbf{B} := \mathrm{diag}(\mathbf{B}_1,\dots,\mathbf{B}_J)$, $\mathbf{e}_b$ denoting the bth column of the adjacency matrix $\mathbf{E}$, and

$$\mathbf{W}_E = (\mathrm{diag}(c_1,\dots,c_J)\otimes \mathbf{I}_p) \times \sum_{b\in\mathcal{B}} \frac{1}{\sum_{\beta\in\mathcal{N}_b} c_\beta}(\mathbf{e}_b\otimes\mathbf{I}_p)(\mathbf{e}_b\otimes\mathbf{I}_p)^T \times (\mathrm{diag}(c_1,\dots,c_J)\otimes\mathbf{I}_p), \tag{35}$$

where $\otimes$ denotes Kronecker product. Substituting (34) into (33), and concatenating $\{\boldsymbol{\phi}_j(k)\}_{j=1}^J$ from (33) in $\boldsymbol{\phi}(k) := [\boldsymbol{\phi}_1^T(k)\dots\boldsymbol{\phi}_J^T(k)]^T$ we obtain (see Appendix F)

$$\boldsymbol{\phi}(k+1) = \hat{\mathbf{x}} + \boldsymbol{\phi}(k) - \mathbf{A}_1\boldsymbol{\phi}(k) - \mathbf{A}_2\boldsymbol{\phi}(k-1) + \bar{\boldsymbol{\eta}}(k) + \bar{\boldsymbol{\eta}}_b(k) \tag{36}$$

where $\hat{\mathbf{x}} := \left[\hat{\mathbf{x}}_1^T,\dots,\hat{\mathbf{x}}_J^T\right]^T$, and the noise vectors $\bar{\boldsymbol{\eta}}(k) := \left[\bar{\boldsymbol{\eta}}_1^T(k)\dots\bar{\boldsymbol{\eta}}_J^T(k)\right]^T$ and $\bar{\boldsymbol{\eta}}_b(k) := \left[\bar{\boldsymbol{\eta}}_{b,1}^T(k)\dots\bar{\boldsymbol{\eta}}_{b,J}^T(k)\right]^T$ have entries

$$\bar{\boldsymbol{\eta}}_j(k) := c_j\mathbf{B}_j^{-1}\sum_{b\in\mathcal{B}_j\setminus\{j\}}\boldsymbol{\eta}_j^b(k)$$
$$\bar{\boldsymbol{\eta}}_{b,j}(k) := c_j\mathbf{B}_j^{-1}\sum_{b\in\mathcal{B}_j}\sum_{j'\in\mathcal{N}_b, j'\neq b}\frac{c_{j'}}{\sum_{\beta\in\mathcal{N}_b}c_\beta}\bar{\boldsymbol{\eta}}_b^{j'}(k). \tag{37}$$

In the convergence analysis of the ensuing section we will need the second-order statistics of $\bar{\boldsymbol{\eta}}(k)$ and $\bar{\boldsymbol{\eta}}_b(k)$. For this reason, we derive in the following lemma their covariance matrices $\boldsymbol{\Sigma}_{\bar{\eta}\bar{\eta}}$ and $\boldsymbol{\Sigma}_{\bar{\eta}_b\bar{\eta}_b}$.

Lemma 5: The noise covariance matrices $\boldsymbol{\Sigma}_{\bar{\eta}\bar{\eta}} := \mathrm{E}\left[\bar{\boldsymbol{\eta}}(k)\bar{\boldsymbol{\eta}}^T(k)\right]$ and $\boldsymbol{\Sigma}_{\bar{\eta}_b\bar{\eta}_b} := \mathrm{E}\left[\bar{\boldsymbol{\eta}}_b(k)\bar{\boldsymbol{\eta}}_b^T(k)\right]$ are

$$\boldsymbol{\Sigma}_{\bar{\eta}\bar{\eta}} = (\mathbf{B}^{-1}(\mathrm{diag}(c_1,\dots,c_J)\otimes\mathbf{I}_p))\mathbf{C}_{\bar{\eta}} \times ((\mathrm{diag}(c_1,\dots,c_J)\otimes\mathbf{I}_p)\mathbf{B}^{-1}) \tag{38}$$
$$\boldsymbol{\Sigma}_{\bar{\eta}_b\bar{\eta}_b} = (\mathbf{B}^{-1}(\mathrm{diag}(c_1,\dots,c_J)\otimes\mathbf{I}_p))\mathbf{C}_{\bar{\eta}_b} \times ((\mathrm{diag}(c_1,\dots,c_J)\otimes\mathbf{I}_p)\mathbf{B}^{-1}) \tag{39}$$

where

(i) the matrix $\mathbf{C}_{\bar{\eta}}$ is formed by $p\times p$ submatrices $[\mathbf{C}_{\bar{\eta}}]_{jj'}$ given by (40) at the bottom of the page.

(ii) the matrix $\mathbf{C}_{\bar{\eta}_b}$ is a $Jp\times Jp$ block diagonal matrix with diagonal blocks $[\mathbf{C}_{\bar{\eta}_b}]_{jj} = \sum_{b\in\mathcal{B}_j\setminus\{j\}}\boldsymbol{\Sigma}_{\eta_j\eta_j}$ for $j=[1,J]$.

Proof: Follows easily from the definitions in (37).

C. Convergence Analysis

The goal of this section is to study the mean and covariance matrix of the difference $\delta\boldsymbol{\phi}(k) := \boldsymbol{\phi}(k)-\boldsymbol{\phi}(k-1)$ in order to establish convergence of $\mathrm{E}[\delta\boldsymbol{\phi}_j(k)]$ to $\hat{\mathbf{s}}_{BL}$ as $k\to\infty$ and bound the covariance matrix $\mathrm{E}\left[\delta\boldsymbol{\phi}_j(k)\delta\boldsymbol{\phi}_j^T(k)\right]$. To this end, let us express $\delta\boldsymbol{\phi}(k)$ as a function of the initial conditions $\boldsymbol{\phi}(0)=\hat{\mathbf{x}}$, $\boldsymbol{\phi}(-1)=\mathbf{0}$, and the noise. This is possible by recursive application of (36) which yields (Appendix G) (41) at the bottom of the page where the matrix $\mathbf{A}\in\mathbb{R}^{2Jp\times 2Jp}$ consists of the $Jp\times Jp$ submatrices $[\mathbf{A}]_{11} = \mathbf{I}_{Jp}-\mathbf{A}_1$, $[\mathbf{A}]_{12} = -\mathbf{A}_2$, $[\mathbf{A}]_{21} = \mathbf{I}_{Jp}$ and $[\mathbf{A}]_{22} = \mathbf{0}_{Jp}$. Since the noise is zero-mean, we have [cf. (41)]

$$\mathrm{E}[\delta\bar{\boldsymbol{\phi}}(k)] = \mathbf{A}^{k-1}\left(\begin{bmatrix}\mathbf{I}-\mathbf{A}_1\\ \mathbf{I}\end{bmatrix}\hat{\mathbf{x}} - \mathbf{A} \times \begin{bmatrix}\mathbf{A}_1 & \mathbf{A}_2\\ -\mathbf{I}_{Jp} & \mathbf{I}_{Jp}\end{bmatrix}\begin{bmatrix}\boldsymbol{\phi}(0)\\ \boldsymbol{\phi}(-1)\end{bmatrix}\right). \tag{42}$$

The covariance matrix during iteration $k+1$ can be easily obtained in terms of $\boldsymbol{\Sigma}_{\bar{\eta}\bar{\eta}_b}$ as [cf. (41)]

$$\begin{aligned}\boldsymbol{\Sigma}_\eta(k+1) :& \\ &= E\left[(\delta\bar{\boldsymbol{\phi}}(k+1) - E[\delta\bar{\boldsymbol{\phi}}(k+1)])\times(\delta\bar{\boldsymbol{\phi}}(k+1) - E[\delta\bar{\boldsymbol{\phi}}(k+1)])^T\right]\\ &= \begin{bmatrix}\mathbf{I}_{Jp} & \mathbf{I}_{Jp}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\boldsymbol{\Sigma}_{\bar{\eta}\bar{\eta}_b}\begin{bmatrix}\mathbf{I}_{Jp} & \mathbf{0}\\ \mathbf{I}_{Jp} & \mathbf{0}\end{bmatrix}\\ &\quad + \sum_{n=0}^{k-1}\mathbf{A}^n\begin{bmatrix}\mathbf{A}_1 & \mathbf{A}_1\\ -\mathbf{I} & -\mathbf{I}\end{bmatrix}\boldsymbol{\Sigma}_{\bar{\eta}\bar{\eta}_b}\begin{bmatrix}\mathbf{A}_1^T & -\mathbf{I}\\ \mathbf{A}_1^T & -\mathbf{I}\end{bmatrix}(\mathbf{A}^T)^n\end{aligned} \tag{43}$$

where $\boldsymbol{\Sigma}_{\bar{\eta}\bar{\eta}_b} = \mathrm{diag}(\boldsymbol{\Sigma}_{\bar{\eta}\bar{\eta}},\boldsymbol{\Sigma}_{\bar{\eta}_b\bar{\eta}_b})$ and $k\geq 1$. Furthermore, let $\lambda_{A,i},\mathbf{u}_{A,i},\mathbf{v}_{A,i}$ denote the ith largest eigenvalue of $\mathbf{A}$ and the corresponding right and left eigenvectors respectively, for

$$[\mathbf{C}_{\bar{\eta}}]_{jj'} = \begin{cases}\displaystyle\sum_{b\in\mathcal{B}_j}\frac{\sum_{j''\in\mathcal{N}_b\setminus\{b\}}c_{j''}^2\boldsymbol{\Sigma}_{\eta_{j''}\eta_{j''}}}{\left(\sum_{\beta\in\mathcal{N}_b}c_\beta\right)^2} & \text{if } j'=j, \text{ and } j=1,\dots,J\\ \displaystyle\sum_{b\in\mathcal{B}_j\cap\mathcal{B}_{j'}}\frac{\sum_{j''\in\mathcal{N}_b\setminus\{b\}}c_{j''}^2\boldsymbol{\Sigma}_{\eta_{j''}\eta_{j''}}}{\left(\sum_{\beta\in\mathcal{N}_b}c_\beta\right)^2} & \text{if } j'\neq j \text{ and } j,j'=1,\dots,J\end{cases} \tag{40}$$

$$\delta\bar{\boldsymbol{\phi}}(k+1) := \begin{bmatrix}\delta\boldsymbol{\phi}(k+1)\\ \delta\boldsymbol{\phi}(k)\end{bmatrix} = \mathbf{A}^{k-1}\left(\begin{bmatrix}\mathbf{I}-\mathbf{A}_1\\ \mathbf{I}\end{bmatrix}\hat{\mathbf{x}} - \mathbf{A}\begin{bmatrix}\mathbf{A}_1 & \mathbf{A}_2\\ -\mathbf{I}_{Jp} & \mathbf{I}_{Jp}\end{bmatrix}\begin{bmatrix}\boldsymbol{\phi}(0)\\ \boldsymbol{\phi}(-1)\end{bmatrix}\right) - \sum_{n=1}^{k}\mathbf{A}^{n-1}\begin{bmatrix}\mathbf{A}_1 & \mathbf{A}_1\\ -\mathbf{I}_{Jp} & -\mathbf{I}_{Jp}\end{bmatrix}\begin{bmatrix}\bar{\boldsymbol{\eta}}(k-n)\\ \bar{\boldsymbol{\eta}}_b(k-n)\end{bmatrix} + \begin{bmatrix}\mathbf{I}_{Jp} & \mathbf{I}_{Jp}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\bar{\boldsymbol{\eta}}(k)\\ \bar{\boldsymbol{\eta}}_b(k)\end{bmatrix} \tag{41}$$

$i = 1, \ldots, 2Jp$. Define also $\bar{\boldsymbol{\Sigma}}_{\bar{\eta}\bar{\eta}_b} := \mathrm{diag}(\boldsymbol{\Sigma}_{\bar{\eta}\bar{\eta}} + \boldsymbol{\Sigma}_{\bar{\eta}_b\bar{\eta}_b}, \mathbf{0}_{Jp})$. Taking limits in (42) and (43) we can characterize the asymptotic behavior of the RD-BLUE algorithm as follows.

Algorithm 3: RD-BLUE

Initialize randomly $\{\boldsymbol{\phi}_j(0) = \hat{\mathbf{x}}, \boldsymbol{\phi}_j(-1) = \mathbf{0}\}_{j=1}^J$.

for $k = 0, 1, \ldots$ **do**

Bridge sensors $b \in \mathcal{B}$: receive $\boldsymbol{\psi}_j(k) + \bar{\boldsymbol{\eta}}_b^j(k)$ from neighbors $j \in \mathcal{N}_b$ and form $\bar{\boldsymbol{\psi}}_b(k)$ using **(34)**;

All $j \in [1, J]$: receive $\bar{\boldsymbol{\psi}}_b(k) + \boldsymbol{\eta}_j^b(k)$ from $b \in \mathcal{B}_j$ to compute $\boldsymbol{\phi}_j(k+1)$ through **(33)**;

All $j \in [1, J]$: Obtain local estimate $\delta\boldsymbol{\phi}_j(k+1) = \boldsymbol{\phi}_j(k+1) - \boldsymbol{\phi}_j(k)$; and transmit $\boldsymbol{\psi}_j(k+1)$ to each $b \in \mathcal{B}_j$;

end for

Proposition 4: The RD-BLUE iterations (33) and (34) reach consensus in the mean sense i.e.

$$\lim_{k\to\infty} E[\delta\boldsymbol{\phi}_j(k)] := \lim_{k\to\infty} E[\boldsymbol{\phi}_j(k) - \boldsymbol{\phi}_j(k-1)] = \hat{\mathbf{s}}_{BL}, \quad j \in [1, J] \tag{44}$$

while the covariance matrix in (43) converges to

$$\lim_{k\to\infty} \boldsymbol{\Sigma}_\eta(k) = \bar{\boldsymbol{\Sigma}}_{\eta\eta_b} + \sum_{i=p+1}^{2Jp} \sum_{i'=p+1}^{2Jp} \frac{\mathbf{u}_{A,i}\mathbf{u}_{A,i'}^T}{1 - \lambda_{A,i}\lambda_{A,i'}} \mathbf{v}_{A,i}^T \times \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_1 \\ -\mathbf{I} & -\mathbf{I} \end{bmatrix} \boldsymbol{\Sigma}_{\bar{\eta}\bar{\eta}_b} \begin{bmatrix} \mathbf{A}_1^T & -\mathbf{I} \\ \mathbf{A}_1^T & -\mathbf{I} \end{bmatrix} \mathbf{v}_{A,i'}. \tag{45}$$

Furthermore, the entries of $\boldsymbol{\Sigma}_\eta(k)$ are bounded.

Proof: See Appendix H.

Proposition 4 establishes convergence of RD-BLUE in the mean. It also shows that even though noise causes the local estimates to fluctuate around the BLUE, their variance remains bounded as $k \to \infty$.

Remark 5: Iterates in the consensus average approach of [19] obey a first-order vector AR process. In order to effect consensus, the largest eigenvalue of the transition matrix defining this AR recursion has to be 1. This entails, alas, an unstable AR process and leads to catastrophic noise propagation. For the coupled oscillators in [1], the consensus is achieved in the derivative of a continuous-time state. Noise resilience is thus expected, and indeed observed in simulations, but not formally established. As per Proposition 4, RD-BLUE is proved to achieve consensus in the mean with local iterates having bounded noise covariance asymptotically quantified by (45).

D. Convergence of CO-BLUE and Comparison

Here we prove the noise resilience of the discretized version of the coupled oscillators (CO) based BLUE in [1]. Upon discretizing the differential equation in [1] we find that the jth sensor receives from all its neighbors $j' \in \mathcal{N}_j$ the noisy vector $\boldsymbol{\phi}_{j'}(k) + \boldsymbol{\eta}_j^{j'}(k)$, and forms $\boldsymbol{\phi}_j(k+1)$ as

$$\boldsymbol{\phi}_j(k+1) = \boldsymbol{\phi}_j(k) + \check{\mathbf{x}}_j + c\mathbf{Q}_j^{-1} \times \sum_{j' \in \mathcal{N}_j \setminus \{j\}} d_{A,jj'}[\boldsymbol{\phi}_{j'}(k) - \boldsymbol{\phi}_j(k)] + c\mathbf{Q}_j^{-1} \times \sum_{j' \in \mathcal{N}_j \setminus \{j\}} d_{A,jj'} \boldsymbol{\eta}_j^{j'}(k), \tag{46}$$

where $\check{\mathbf{x}}_j := \left(\mathbf{H}_j^T \boldsymbol{\Sigma}_{x_j x_j}^{-1} \mathbf{H}_j\right)^{-1} \mathbf{H}_j \boldsymbol{\Sigma}_{x_j x_j}^{-1} \mathbf{x}_j$, $\mathbf{Q}_j := \mathbf{H}_j^T \boldsymbol{\Sigma}_{x_j x_j}^{-1} \mathbf{H}_j$, and $d_{A,jj'} = d_{A,j'j} \geq 0$ is a weight associated with the communication link between sensors j and j' so that if $j' \notin \mathcal{N}_j$ then $d_{A,jj'} = 0$. Unlike $\hat{\mathbf{x}}_j$, note that $\check{\mathbf{x}}_j$ does not incorporate a regularization term. Furthermore, CO-BLUE does not use bridge sensors, which explains the usage of j and j' (instead of j and b). The noise vector $\boldsymbol{\eta}_j^{j'}(k)$ has zero mean and covariance matrix $\boldsymbol{\Sigma}_{\eta_j \eta_j}$, and is uncorrelated across sensors and time. Furthermore, let $\mathbf{Z} := \mathrm{diag}\left(\mathbf{Q}_1^{-1}, \ldots, \mathbf{Q}_J^{-1}\right) \mathbf{P}^T \left(\mathbf{I}_p \otimes \boldsymbol{\Xi}\mathbf{D}_A\boldsymbol{\Xi}^T\right) \mathbf{P}$, where $\mathbf{P}$ is a permutation matrix whose structure is detailed in [1], while $\boldsymbol{\Xi} \in \mathbb{R}^{J \times N_e}$ is the incidence matrix of the communication graph for which $[\boldsymbol{\Xi}]_{ji} = 1$ if edge i is incoming to sensor j, $[\boldsymbol{\Xi}]_{ji} = -1$ if the edge is outgoing, and zero otherwise; $\mathbf{D}_A$ is a diagonal matrix with diagonal entries $d_{A,jj'}$ and N_e denotes the total number of edges.

Concatenating (46) $\forall\, j \in [1, J]$, the global CO-BLUE recursion in the presence of noise is

$$\boldsymbol{\phi}(k+1) = \check{\mathbf{x}}' + (\mathbf{I} - c\mathbf{Z})\boldsymbol{\phi}(k) + \bar{\boldsymbol{\eta}}'(k) \tag{47}$$

where $\bar{\boldsymbol{\eta}}'(k)$ contains all the noise summands in (46), while $\check{\mathbf{x}} := \left[\check{\mathbf{x}}_1^T \ldots \check{\mathbf{x}}_J^T\right]^T$. Using the steps in Section V-C we can also establish the noise-robustness of CO-BLUE; i.e.

$$\lim_{k\to\infty} E[\delta\boldsymbol{\phi}_j(k)] = \lim_{k\to\infty} E[\boldsymbol{\phi}_j(k) - \boldsymbol{\phi}_j(k-1)] = \hat{\mathbf{s}}_{BL} \tag{48}$$

$$\lim_{k\to\infty} \boldsymbol{\Sigma}_{\eta'}(k) = \boldsymbol{\Sigma}_{\bar{\eta}'\bar{\eta}'} + \sum_{i=p+1}^{2Jp} \sum_{i'=p+1}^{2Jp} \frac{\mathbf{u}_{z,i}\mathbf{u}_{z,i'}^T}{1 - (1 - c\lambda_{Z,i})(1 - c\lambda_{Z,i'})} \times \mathbf{u}_{z,i}^T c\mathbf{Z}\boldsymbol{\Sigma}_{\bar{\eta}'\bar{\eta}'} c\mathbf{Z}\mathbf{u}_{z,i} \tag{49}$$

where $\boldsymbol{\Sigma}_{\bar{\eta}'\bar{\eta}'}$ is the covariance matrix of $\bar{\boldsymbol{\eta}}'(k)$, and $\{\lambda_{Z,i}, \mathbf{u}_{z,i}\}_{i=1}^p$ are the eigenpairs of matrix $\mathbf{Z}$. From (47) it can be seen that CO-BLUE also achieves consensus in the mean across the WSN so long as $0 < c < 2(\max_{i \in [1,Jp]} |\lambda_{Z,i}|)^{-1}$.

Remark 6: It is apparent from (41) that the convergence rate of RD-BLUE can be adjusted by the penalty coefficients $\{c_j\}_{j=1}^J$, while in CO-BLUE this can be done through the coefficient c. Thus, RD-BLUE has J degrees of freedom for adjusting the convergence speed, while CO-BLUE has only one. Furthermore, notice that c_j's in RD-BLUE can assume any positive value, while c in CO-BLUE must be restricted in the interval $(0, 2(\max_{i \in [1,Jp]} |\lambda_{Z,i}|)^{-1})$ to ensure convergence. These features explain why RD-BLUE is more flexible than CO-BLUE in trading-off convergence speed for steady-state error.

E. Simulations

Here we test the convergence of D-BLUE and RD-BLUE, and compare them with the CO-BLUE in [1] and the consensus average (CA) BLUE in [20]. Furthermore, we examine the noise resilience properties of the aforementioned algorithms in the presence of either reception or quantization noise. We use the same WSN with $J = 60$ sensors as in Section III-D where the jth sensor observations obey $\mathbf{x}_j = \mathbf{H}_j\mathbf{s} + \mathbf{n}_j, \forall\, j$. In Fig. 3 (top) we consider ideal channel links and plot the normalized error $e_{\text{norm}}(k)$ in (19) versus k for D-BLUE, RD-BLUE, CO-BLUE, and CA-BLUE. For the D-BLUE and RD-BLUE algorithms, the penalty coefficients are set to $c_j = 10/|\mathcal{B}_j|$. For the CO-BLUE

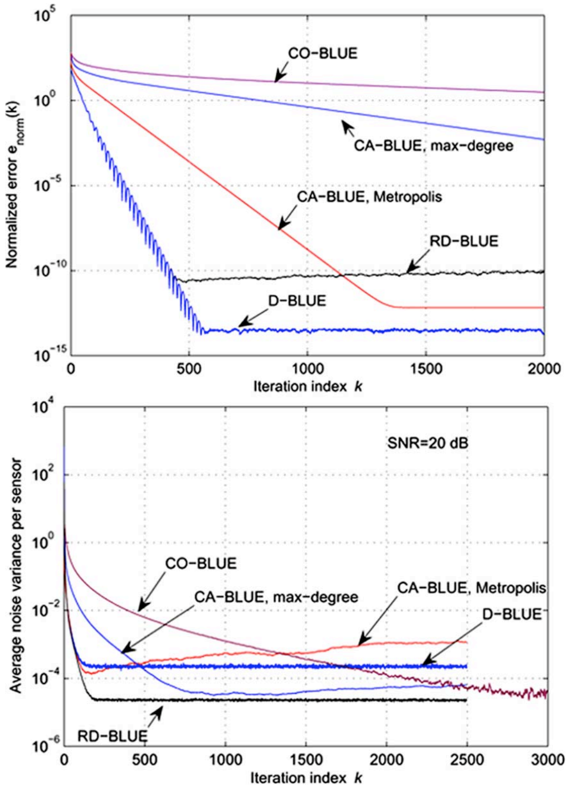

Fig. 3. Normalized error $e_{\mathrm{norm}}(k)$ versus k for D-BLUE, RD-BLUE, CA-BLUE and CO-BLUE under ideal channel links (top); Average noise variance per sensor versus k in the presence of reception noise with $\mathrm{SNR} = 20$ dB (bottom).

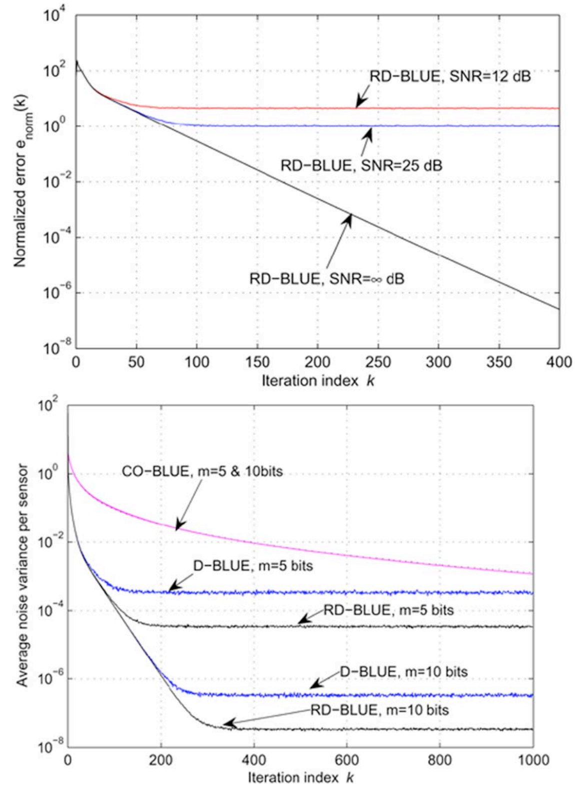

Fig. 4. Normalized error $e_{\mathrm{norm}}(k)$ versus k for RD-BLUE in the presence of reception noise with $\mathrm{SNR} = 12, 25$ and ∞ dB (top). Average noise variance per sensor versus k for RD-BLUE and CO-BLUE in the presence of quantization noise when using $m = 5$, 10 and ∞ bits per observation (bottom).

algorithm, we set $c = c^*$ where c^* is the optimal value yielding the highest possible convergence rate. Specifying c^* requires global network information, see, e.g., [18]; hence, this choice is the best case scenario for [1]. Also, for CA-BLUE we adopt the max-degree and Metropolis weights, which allow for distributed implementation as in [20, eqs. (8) and (9)]. Clearly, Fig. 3 (top) demonstrates that D-BLUE and RD-BLUE attain higher convergence rates, outperforming both CA-BLUE and CO-BLUE. The price paid for higher convergence speed is a slightly higher steady-state error in RD-BLUE.

Fig. 3 (bottom) displays the average noise variance per sensor, namely $\mathrm{trace}(\boldsymbol{\Sigma}_\eta(k))/J$, versus iteration index k, after incorporating reception noise in the sensor links so that $\mathrm{SNR} = 20$ dB. Specifically, the noise variance per sensor is computed via ensemble averaging across sensors and across 50 different realizations of the RD-BLUE, D-BLUE, CO-BLUE and CA-BLUE algorithms. For a fair comparison between RD-BLUE and CO-BLUE we set the $c_j = 1/|\mathcal{B}_j|$ for $j = 1, \ldots, 60$ and $c = c^*$ such that the steady-state noise variance is $\mathrm{trace}(\lim_{k\to\infty} \boldsymbol{\Sigma}_\eta(k)) = 1.4 \times 10^{-3}$, which amounts to an average noise variance of 2.33×10^{-5} per sensor. The penalty coefficients for D-BLUE are set as in RD-BLUE. As expected, CA-BLUE eventually diverges in the presence of noise. Notice though that similar to D-MLE the D-BLUE algorithm exhibits noise resilience, at the expense of a higher steady-state variance than RD-BLUE. But RD-BLUE achieves higher convergence rate than CO-BLUE while the steady-state error variance is the same for both schemes. Thus, RD-BLUE is flexible to tradeoff convergence rate for steady-state error variance.

In Fig. 4 (top), we plot $e_{\mathrm{norm}}(k)$ for different reception SNRs. Clearly, as the SNR increases the steady-state error reduces, while for $\mathrm{SNR} = \infty$ all sensors converge to the BLUE. The same behavior is also displayed by RD-BLUE in Fig. 4 (bottom), which depicts the average noise variance per sensor versus k, for a variable number of quantization bits (common across all sensors). Furthermore, the convergence rate of RD-BLUE is higher than that of CO-BLUE for the same steady-state noise variance.

In Fig. 5, we test the convergence of RD-BLUE for different coefficients c_j under ideal channel links, and compare it with CO-BLUE for $c = c^*$. Simulations indicate that a proper selection of c_j is $c_j = a_j/|\mathcal{B}_j|$, where a_j is a positive real scalar. Intuitively, these coefficients weigh evenly the cost function and the constraints in the augmented Lagrangian in (22). Observe that a reasonable selection for a_j is to set it equal to $a_j = 4L_j = 20$. Fig. 5 indicates that proper selection of c_j leads to high convergence speeds, enabling D-BLUE and RD-BLUE to outperform CA-BLUE and CO-BLUE (cf. Fig. 3 (top) and Fig. 5).

VI. Concluding Remarks

We developed distributed algorithms for estimation of unknown deterministic signals using *ad hoc* WSNs based on successive refinement of local estimates. The crux of our approach is to express the desired estimator, either MLE or BLUE, as the solution of judiciously formulated convex optimization problems. We then used the method of multipliers combined with block coordinate descent updates to enable decentralized implementation. Our methodology does not require the estimator to be known as a closed-form expression in terms of averages, while it allows development of distributed algorithms, even for

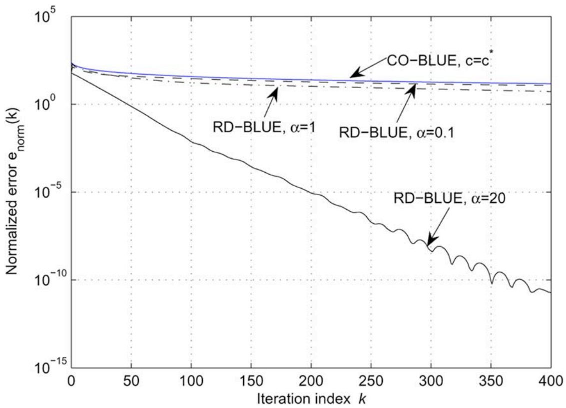

Fig. 5. Normalized error $e_{\rm norm}(k)$ versus k for RD-BLUE and CO-BLUE in ideal links and for different penalty coefficients c_j's.

nonlinear estimators. In fact, the formulation and resultant algorithms apply even without any random considerations to linear and nonlinear least-squares fit problems since the latter can be viewed as BLUE and MLE problems when the data model is linear and the unmodeled dynamics are assumed Gaussian.

Furthermore, our schemes exhibit resilience to communication and/or quantization noise. When it comes to decentralized computation of linear estimators, namely the BLUE, we constructed noise-robust algorithms whose convergence can be analyzed and quantified through the covariance structure of the noise contaminating the local estimates. Different from the consensus averaging approach, noise covariance in RD-BLUE converges to a bounded matrix corroborating its noise resilient features.

Through the D-BLUE and RD-BLUE algorithms improved convergence rates were possible for uncorrelated observations across space. Ongoing research to be reported in part II of this work considers our optimization framework in developing decentralized estimation algorithms with correlated observations even for random signals, where focus is placed on distributed computation of the linear minimum mean-square error estimator. The same estimation task will also be considered in dynamic and nonlinear setups. Additional future research directions include comparisons with spanning tree and gossip type algorithms on the basis of convergence speed and power consumption[2].

APPENDIX

A. Proof of Proposition 1

We will show first that the constraints $\mathbf{s}_j = \bar{\mathbf{s}}_b$ for $b \in \mathcal{B}$ and $j \in \mathcal{N}_b$ are equivalent to $\mathbf{s}_1 = \cdots = \mathbf{s}_J$. To this end, consider $b_1, b_2 \in \mathcal{B}$ with $b_1 \in \mathcal{N}_{j_1}$ and $b_2 \in \mathcal{N}_{j_2}$, with the existence of b_1, b_2 guaranteed by Definition 1. From the constraints in (4) we know that

$$\mathbf{s}_{j_i} = \bar{\mathbf{s}}_{b_i}, \text{ for } i = 1, 2. \tag{50}$$

On the other hand, (a1) guarantees existence of a path $\mathcal{P}$ connecting $b_1, b_2 \in \mathcal{B}$. Moreover, from Def. 1-(a,b) we know that every sensor $i \in \mathcal{P}$ must have at least two neighbors $b'_1, b'_2 \in \mathcal{B} \cap \mathcal{P}$, otherwise there would be sensors in $\mathcal{P} \cap \mathcal{B}$ for which there is no sensor in $\mathcal{B}$ at most 2 edges away from them. We can thus build a path from b_1 to b_2 of the form $b_1 \to i_1 \to b'_1 \to i_2 \to b'_2 \to \cdots \to b'_K \to i_K \to b_2$, for which $\bar{\mathbf{s}}_{b_1} = \mathbf{s}_{i_1} = \bar{\mathbf{s}}_{b'_1} = \cdots = \bar{\mathbf{s}}_{b'_K} = \mathbf{s}_{i_K} = \bar{\mathbf{s}}_{b_2}$. Combining the latter with (50), it follows that $\mathbf{s}_{j_1} = \mathbf{s}_{j_2}$ for arbitrary $j_1, j_2 \in [1, J]$. Thus, any feasible point of (4) satisfies $\mathbf{s}_j = \mathbf{s}$, for all $j \in [1, J]$ implying that the arguments of (1) and (4) are equal, which completes the proof. □

B. Proof of Proposition 2

With $\mathcal{B}$ denoting a bridge sensor subset, we wish to show that (7)–(9) generate a series of local estimates converging to the optimal solution of (4), namely the MLE estimator. We will establish this by showing that (7)–(9) correspond to the steps involved in the alternating-direction method of multipliers [2, pg. 255]. To this end, let $\mathbf{v}_j^b(k)$ denote the Lagrange multipliers at the kth iteration. Moreover, define $\mathbf{F} := \left[\mathbf{F}_1^T \ldots \mathbf{F}_{|B|}^T\right]^T$ of size $(\sum_{b\in\mathcal{B}} |\mathcal{N}_b|)p \times Jp$, where $\mathbf{F}_b := \left[\boldsymbol{\epsilon}_{b_1}^T \ldots \boldsymbol{\epsilon}_{b_{|\mathcal{N}_b|}}^T\right]^T \otimes \mathbf{I}_p$ and $\boldsymbol{\epsilon}_{b_1} \in \mathbb{R}^{J\times 1}$ denotes the vector with b_1th entry one and zero elsewhere, while $b_1 > b_2 > \cdots > b_{|\mathcal{N}_b|}$ are the indices of the nonzero entries in the bth column of $\mathbf{E}$. Then, (4) can be equivalently written as

$$\{\hat{\mathbf{s}}_j\}_{j=1}^J = \arg\min G_1(\boldsymbol{s}) + G_2(\mathbf{F}\boldsymbol{s}), \text{ s. to } \bar{\boldsymbol{s}} := \mathbf{F}\boldsymbol{s} \in \mathcal{C} \tag{51}$$

where $G_1(\boldsymbol{s}) := -\sum_{j=1}^J \ln[p_j(\mathbf{x}_j; \mathbf{s}_j)]$, $G_2(\bar{\boldsymbol{s}}) = 0$, while $\boldsymbol{s} := \left[\mathbf{s}_1^T \ldots \mathbf{s}_J^T\right]^T$, and $\mathcal{C} \subset \mathbb{R}^{\sum_{b\in\mathcal{B}} |N_b|(p)\times 1}$ is the polyhedral set defined so that for $\mathbf{F}_b\boldsymbol{s} := \left[\left(\bar{\mathbf{s}}_b^1\right)^T \ldots \left(\bar{\mathbf{s}}_b^{|\mathcal{N}_b|}\right)^T\right]^T$ it holds that $\bar{\mathbf{s}}_b^1 = \cdots = (\bar{\mathbf{s}}_b^{|\mathcal{N}_b|}) = \bar{\mathbf{s}}_b$ for all $b \in \mathcal{B}$. Inspection of (51) shows that it has the same form as the optimization problem in [2, Eq. (4.76)]. Thus, the steps of the alternating-direction method of multipliers at $(k+1)$-st iteration are:

[S1] Set $\mathbf{v} := \mathbf{v}(k) = \left\{\mathbf{v}_j^b(k)\right\}_{j\in[1,J]}^{b\in\mathcal{B}_j}$ and $\{\bar{\mathbf{s}}_b = \bar{\mathbf{s}}_b(k)\}_{b\in\mathcal{B}}$ to obtain $\boldsymbol{s}(k+1)$ by solving the following minimization problem

$$[\boldsymbol{s}(k+1)] = \arg\min_{\boldsymbol{s}} \mathcal{L}_a[\boldsymbol{s}, \{\bar{\mathbf{s}}_b(k)\}_{b\in\mathcal{B}}, \mathbf{v}(k)] \tag{52}$$

[S2] For fixed $\mathbf{v} := \mathbf{v}(k) = \left\{\mathbf{v}_j^b(k)\right\}_{j\in[1,J]}^{b\in\mathcal{B}_j}$, and setting $\boldsymbol{s} = \boldsymbol{s}(k+1)$ after completing step [S1], the consensus variables $\bar{\mathbf{s}}_b(k+1)$ for $b \in \mathcal{B}$ are obtained as

$$\begin{aligned}[\{\bar{\mathbf{s}}_b(k+1)\}_{b\in\mathcal{B}}] = \arg\min_{\bar{\boldsymbol{s}}} \Bigg(&-\sum_{b\in\mathcal{B}}\sum_{j\in\mathcal{N}_b} \left(\mathbf{v}_j^b(k)\right)^T \bar{\mathbf{s}}_b \\ &+ \sum_{b\in\mathcal{B}}\sum_{j\in\mathcal{N}_b} \frac{c_j}{2}\|\mathbf{s}_j(k+1) - \bar{\mathbf{s}}_b\|_2^2\Bigg)\end{aligned} \tag{53}$$

[S3] Update $\left\{\mathbf{v}_j^b(k)\right\}_{j\in[1,J]}^{b\in\mathcal{B}_j}$ via (7).

Utilizing (6), we infer that (52) is equivalent to the following separate J sub-problems

$$\begin{aligned}\mathbf{s}_j(k+1) = \arg\min_{\mathbf{s}_j} \Bigg(&-\ln[p_j(\mathbf{x}_j; \mathbf{s}_j)] + \sum_{b\in\mathcal{B}_j} \left(\mathbf{v}_j^b(k)\right)^T \mathbf{s}_j \\ &+ \sum_{j\in\mathcal{B}_j} \frac{c_j}{2}\|\mathbf{s}_j - \bar{\mathbf{s}}_b(k)\|_2^2\Bigg).\end{aligned} \tag{54}$$

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

Similarly, $\bar{\mathbf{s}}_b(k+1)$ can be obtained by minimizing the cost function formed by keeping only the bth term of the outer sums in (53). Interestingly, each of the optimization problems in (54) can be solved locally at the corresponding sensor. Notice that (54) coincides with (8). Further, setting the gradient of the cost function formed by the bth summand (of the outer sum) in (53), with respect to $\bar{\mathbf{s}}_b$, equal to zero we obtain (9). We have shown that the alternating direction method of multipliers applied to (1), boils down to (7)–(9). Since (1) is convex and $\mathbf{F}^T\mathbf{F}$ is invertible, recursions (7)–(9) converge to the optimal solution $\hat{\mathbf{s}}_{BL}$ [2, pg. 257–260]. □

C. Proof of Lemma 1

The cost in (20), call it $F_{BL}(\mathbf{s})$, is convex. Its optimal solution can thus be obtained by applying the first-order optimality conditions. Specifically, the gradient of $F_{BL}(\mathbf{s})$ is

$$\nabla_{\mathbf{s}} F_{BL}(\mathbf{s}) = 2\sum_{j=1}^{J} \mathbf{H}_j^T \mathbf{\Sigma}_{x_j x_j}^{-1} \mathbf{H}_j \mathbf{s} - 2\sum_{j=1}^{J} \mathbf{H}_j^T \mathbf{\Sigma}_{x_j x_j}^{-1} \mathbf{x}_j. \quad (55)$$

Setting $\nabla_{\mathbf{s}} F_{BL}(\mathbf{s}) = \mathbf{0}$, we find that the optimal solution of (20) coincides with $\hat{\mathbf{s}}_{BL}$ in (2). □

D. Proof of Lemma 3

With initial conditions $\left\{\mathbf{v}_j^b(-1) = \mathbf{0}\right\}_{j\in[1,J]}^{b\in\mathcal{B}_j}$, (25) establishes that (28) holds true for $k = -1$. Next, arguing by induction let us assume that $\bar{\mathbf{s}}_b(n)$ is given by (28) for $n \le k$. Substituting successively $\mathbf{v}_j^b(k)$ in (23), we arrive at

$$\mathbf{v}_j^b(k+1) = \mathbf{v}_j^b(-1) + c_j \sum_{n=1}^{k+1} (\mathbf{s}_j(n) - \bar{\mathbf{s}}_b(n)) \qquad b \in \mathcal{B}_j, \quad j \in [1, J]. \quad (56)$$

Substituting (56) into (25), while setting $\mathbf{v}_j^b(-1) = \mathbf{0}$, we obtain (28) for $k \ge 0$ using:

$$\begin{aligned} \bar{\mathbf{s}}_b(k+1) &= \frac{1}{\sum_{\beta\in\mathcal{N}_b} c_\beta} \sum_{j\in\mathcal{N}_b} \left[c_j \sum_{n=1}^{k} \mathbf{s}_j(n) - c_j \sum_{n=1}^{k} \bar{\mathbf{s}}_b(n) \right] + \sum_{j\in\mathcal{N}_b} \frac{c_j}{\sum_{\beta\in\mathcal{N}_b} c_\beta} \mathbf{s}_j(k+1) \\ &= \frac{1}{\sum_{\beta\in\mathcal{N}_b} c_\beta} \sum_{j\in\mathcal{N}_b} c_j \sum_{n=1}^{k+1} \mathbf{s}_j(n) - \sum_{n=1}^{k} \bar{\mathbf{s}}_b(n) \\ &= \sum_{j\in\mathcal{N}_b} \frac{c_j}{\sum_{\beta\in\mathcal{N}_b} c_\beta} \mathbf{s}_j(k+1). \end{aligned} \quad (57)$$

Combining (57) with (24), (27) follows easily since

$$\begin{aligned} \mathbf{s}_j(k+1) &= \mathbf{B}_j^{-1} \left[2\mathbf{H}_j^T \mathbf{\Sigma}_{x_j x_j}^{-1} \mathbf{x}_j - \sum_{b\in\mathcal{B}_j} \sum_{n=1}^{k} c_j (\mathbf{s}_j(n) - \bar{\mathbf{s}}_b(n)) + c_j \sum_{b\in\mathcal{B}_j} \bar{\mathbf{s}}_b(k) \right] \\ &= \mathbf{s}_j(k) - |\mathcal{B}_j| c_j \mathbf{B}_j^{-1} \mathbf{s}_j(k) + \mathbf{B}_j^{-1} c_j \sum_{b\in\mathcal{B}_j} (2\bar{\mathbf{s}}_b(k) - \bar{\mathbf{s}}_b(k-1)). \end{aligned}$$

From (24) and after setting $\mathbf{v}_j^b(-1) = \mathbf{0}$ and $\bar{\mathbf{s}}_b(-1) = \mathbf{0}$, it follows that $\mathbf{s}_j(0) = \hat{\mathbf{x}}_j$ for $j = 1, \ldots, J$. □

E. Proof of Lemma 4

Using the initial conditions $\boldsymbol{\phi}_j(0) = \hat{\mathbf{x}}_j$, $\boldsymbol{\phi}_j(-1) = \mathbf{0}$ and $\boldsymbol{\phi}_j(-2) = \mathbf{0}$ we obtain that $\mathbf{s}_j(-1) = \mathbf{0} = \boldsymbol{\phi}_j(-1) - \boldsymbol{\phi}_j(-2)$, and $\mathbf{s}_j(0) = \hat{\mathbf{x}}_j = \boldsymbol{\phi}_j(0) - \boldsymbol{\phi}_j(-1)$. Thus, Lemma 4 holds true for $k = -1, 0$. Next, using (29) we obtain the recursions for $\boldsymbol{\phi}_j(k+1)$ and $\boldsymbol{\phi}_j(k)$ and subtract the second from the first, while we replace the $\bar{\boldsymbol{\phi}}_b$ terms by (30) to arrive at

$$\begin{aligned} &\boldsymbol{\phi}_j(k+1) - \boldsymbol{\phi}_j(k) \\ &= c_j \mathbf{B}_j^{-1} \sum_{b\in\mathcal{B}_j} \left[\sum_{j\in\mathcal{N}_b} \frac{c_j}{\sum_\beta c_\beta} [2(\boldsymbol{\phi}_j(k) - \boldsymbol{\phi}_j(k-1)) - (\boldsymbol{\phi}_j(k-1) - \boldsymbol{\phi}_j(k-2))] \right] \\ &\quad + \left(\mathbf{I} - c_j |\mathcal{B}_j| \mathbf{B}_j^{-1}\right) (\boldsymbol{\phi}_j(k) - \boldsymbol{\phi}_j(k-1)). \end{aligned} \quad (58)$$

From (58) and after recalling that $\boldsymbol{\phi}_j(-1) - \boldsymbol{\phi}_j(-2) = \mathbf{s}_j(-1) = \mathbf{0}$ and $\boldsymbol{\phi}_j(0) - \boldsymbol{\phi}_j(-1) = \mathbf{s}_j(0) = \hat{\mathbf{x}}_j$, it follows immediately that the iterates $\boldsymbol{\phi}_j(k+1) - \boldsymbol{\phi}_j(k)$ are equal to $\mathbf{s}_j(k+1)$ in (27) for $k \ge -1$ and subsequently the iterates $\bar{\boldsymbol{\phi}}_b(k+1) - \bar{\boldsymbol{\phi}}_b(k)$ have the same value with $\bar{\mathbf{s}}_b(k+1)$ in (28). □

F. Proof of (36)

From (33) and (34) we obtain the following:

$$\begin{aligned} \boldsymbol{\phi}_j(k+1) = &\hat{\mathbf{x}}_j + \left(\mathbf{I}_p - c_j |\mathcal{B}_j| \mathbf{B}_j^{-1}\right) \boldsymbol{\phi}_j(k) + 2c_j \mathbf{B}_j^{-1} \\ &\times \sum_{b\in\mathcal{B}_j} \sum_{j'\in\mathcal{N}_b} \frac{c_{j'} \boldsymbol{\phi}_{j'}(k)}{\sum_{\beta\in\mathcal{N}_b} c_\beta} - c_j \mathbf{B}_j^{-1} \\ &\times \sum_{b\in\mathcal{B}_j} \sum_{j'\in\mathcal{N}_b} \frac{c_{j'} \boldsymbol{\phi}_{j'}(k-1)}{\sum_{\beta\in\mathcal{N}_b} c_\beta} \\ &+ c_j \mathbf{B}_j^{-1} \sum_{b\in\mathcal{B}_j\setminus\{j\}} \boldsymbol{\eta}_j^b(k) + c_j \mathbf{B}_j^{-1} \\ &\times \sum_{b\in\mathcal{B}_j} \sum_{j'\in\mathcal{N}_b, j'\neq b} \frac{c_{j'}}{\sum_{\beta\in\mathcal{N}_b} c_\beta} \bar{\boldsymbol{\eta}}_b^{j'}(k). \end{aligned} \quad (59)$$

The first two terms in (36) as well as the first term in $\mathbf{A}_1\boldsymbol{\phi}(k)$ can be obtained readily by stacking the first two terms in (59). Similarly the noise terms $\bar{\boldsymbol{\eta}}(k)$ and $\bar{\boldsymbol{\eta}}_b(k)$ can be obtained by stacking the last two terms in (59). Here, we show how to obtain the second matrix term in $\mathbf{A}_1$, as well as $\mathbf{A}_2$. Toward this end, we can express the third term in (59) as follows (we exclude the $2c_j\mathbf{B}_j^{-1}$ term):

$$\begin{aligned} \sum_{b\in\mathcal{B}_j} \sum_{j'\in\mathcal{N}_b} \frac{c_{j'} \boldsymbol{\phi}_{j'}(k)}{\sum_{\beta\in\mathcal{N}_b} c_\beta} &= \sum_{b\in\mathcal{B}} \mathbf{E}_{jb} \mathbf{I}_p \sum_{j'\in\mathcal{N}_b} \frac{c_{j'} \boldsymbol{\phi}_{j'}(k)}{\sum_{\beta\in\mathcal{N}_b} c_\beta} \\ &= \sum_{b\in\mathcal{B}} \frac{\mathbf{E}_{jb}}{\sum_{\beta\in\mathcal{N}_b} c_\beta} \mathbf{I}_p \left(\mathbf{e}_b^T \otimes \mathbf{I}_p\right) \times \left[c_1 \boldsymbol{\phi}_1{}^T(k) \ldots c_J \boldsymbol{\phi}_J{}^T(k)\right]^T \\ &= \sum_{b\in\mathcal{B}} \frac{\mathbf{E}_{jb}}{\sum_{\beta\in\mathcal{N}_b} c_\beta} \mathbf{I}_p (\mathbf{e}_b \otimes \mathbf{I}_p)^T \times (\mathrm{diag}(c_1, \ldots, c_J) \otimes \mathbf{I}_p) \boldsymbol{\phi}(k). \end{aligned} \quad (60)$$

Stacking the third term in (59) $\forall\, j \in [1, J]$ in a column vector, namely $\mathbf{y}_a(k)$, and using (60) we obtain

$$\begin{aligned} \mathbf{y}_a(k) = &-2\mathbf{B}^{-1} (\mathrm{diag}(c_1, \ldots, c_J) \otimes \mathbf{I}_p) \\ &\times \sum_{b\in\mathcal{B}} \frac{\mathbf{e}_b \otimes \mathbf{I}_p}{\sum_{\beta\in\mathcal{N}_b} c_\beta} (\mathbf{e}_b \otimes \mathbf{I}_p)^T (\mathrm{diag}(c_1, \ldots, c_J) \otimes \mathbf{I}_p) \boldsymbol{\phi}(k), \end{aligned} \quad (61)$$

from which we readily obtain that $\mathbf{y}_a(k) = -2\mathbf{B}^{-1}\mathbf{W}_E\boldsymbol{\phi}(k)$. Following the same approach we can derive the fourth term in (36). □

G. Proof of (41)

Using induction we first show that (41) holds true for $k=1$. After substituting $k=1$ in (41) and using (41) we obtain after easy manipulations that $\delta\boldsymbol{\phi}(2) = \hat{\mathbf{x}} - \mathbf{A}_1\boldsymbol{\phi}(1) - \mathbf{A}_2\boldsymbol{\phi}(0) + \bar{\boldsymbol{\eta}}(1) + \bar{\boldsymbol{\eta}}_b(1) = \boldsymbol{\phi}(2) - \boldsymbol{\phi}(1)$, and $\delta\boldsymbol{\phi}(1) = \hat{\mathbf{x}} - \mathbf{A}_1\boldsymbol{\phi}(0) - \mathbf{A}_2\boldsymbol{\phi}(-1) + \bar{\boldsymbol{\eta}}(0) + \bar{\boldsymbol{\eta}}_b(0)$, which both hold true from (36). Next, assuming that (41) holds true for k we show that the same applies for $k+1$. To this end, recall that $\boldsymbol{\phi}(k) = \boldsymbol{\phi}(k-1) + \delta\boldsymbol{\phi}(k)$, and let $\boldsymbol{\phi}_2(k+1) = [\boldsymbol{\phi}^T(k+1)\boldsymbol{\phi}^T(k)]^T$. Then, we have

$$\begin{aligned}
\boldsymbol{\phi}_2(k+1) &\\
&= \mathbf{A}\boldsymbol{\phi}_2(k) + [\hat{\mathbf{x}}^T\mathbf{0}]^T + \left[\bar{\boldsymbol{\eta}}^T(k) + \bar{\boldsymbol{\eta}}_b^T(k)\mathbf{0}^T\right]^T \\
&= \mathbf{A}\boldsymbol{\phi}_2(k-1) + \begin{bmatrix}\hat{\mathbf{x}}\\ \mathbf{0}\end{bmatrix} + \begin{bmatrix}\bar{\boldsymbol{\eta}}(k-1) + \bar{\boldsymbol{\eta}}_b(k-1)\\ \mathbf{0}\end{bmatrix} \\
&\quad + \mathbf{A}^{k-1}\left(\begin{bmatrix}\mathbf{I}-\mathbf{A}_1\\ \mathbf{I}\end{bmatrix}\hat{\mathbf{x}} - \mathbf{A}\begin{bmatrix}\mathbf{A}_1 & \mathbf{A}_2\\ -\mathbf{I}_{Jp} & \mathbf{I}_{Jp}\end{bmatrix}\begin{bmatrix}\boldsymbol{\phi}(0)\\ \boldsymbol{\phi}(-1)\end{bmatrix}\right) \\
&\quad - \sum_{n=1}^{k}\mathbf{A}^{n-1}\begin{bmatrix}\mathbf{A}_1 & \mathbf{A}_1\\ -\mathbf{I}_{Jp} & -\mathbf{I}_{Jp}\end{bmatrix}\begin{bmatrix}\bar{\boldsymbol{\eta}}(k-n)\\ \bar{\boldsymbol{\eta}}_b(k-n)\end{bmatrix} \\
&\quad + \begin{bmatrix}\mathbf{I}_{Jp} & \mathbf{I}_{Jp}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\bar{\boldsymbol{\eta}}(k)\\ \bar{\boldsymbol{\eta}}_b(k)\end{bmatrix} \qquad (62)
\end{aligned}$$

where the second equality in (62) is derived after utilizing (41) to expand $\boldsymbol{\phi}_2(k)$. Upon recognizing that the sum of the first three summands in (62) is equal to $\boldsymbol{\phi}_2(k)$ and subtracting this vector from the rhs of (62), it follows easily that $\delta\bar{\boldsymbol{\phi}}(k+1)$ is given by (41). □

H. Proof of Proposition 4

First we will establish properties of matrix $\mathbf{A}$ which will be used in the convergence analysis of RD-BLUE. These properties are summarize next:

Lemma 6: The eigenvalues $\{\lambda_{A,i}\}_{i=1}^{2Jp}$ of $\mathbf{A}$ ordered so that $|\lambda_{A,1}| \geq \cdots \geq |\lambda_{A,2Jp}|$ and the corresponding right and left eigenvectors $\mathbf{u}_{A,i}$ and $\mathbf{v}_{A,i}$ satisfy the following properties:

(a) It holds that $|\lambda_{A,i}| = 1$ for $i = 1,\ldots,p$; while $|\lambda_{A,i}| < 1$ for $i \in [p+1, 2Jp]$.

(b) The p dominant right eigenvectors $\{\mathbf{u}_{A,i}\}_{i=1}^{p}$ have the form $\mathbf{u}_{A,i} = \left[\boldsymbol{\epsilon}_i^T \ldots \boldsymbol{\epsilon}_i^T\right]^T \in \mathbb{R}^{2Jp\times 1}$, where $\boldsymbol{\epsilon}_i \in \mathbb{R}^{p\times 1}$ denotes the vector having one in its ith entry and zeros elsewhere.

(c) The p dominant left eigenvectors $\mathbf{v}_{A,i} := \left[\left(\mathbf{v}_{A,i}^1\right)^T \left(\mathbf{v}_{A,i}^2\right)^T\right]^T$ are given by:

$$\begin{aligned}
\left(\mathbf{v}_{A,i}^1\right)^T &= \frac{\boldsymbol{\epsilon}_i^T}{2}\left(\sum_{j=1}^{J}\mathbf{H}_j^T\boldsymbol{\Sigma}_{x_jx_j}^{-1}\mathbf{H}_j\right)^{-1}[\mathbf{B}_1\ldots\mathbf{B}_J] \in \mathbb{R}^{Jp\times 1}\\
\left(\mathbf{v}_{A,i}^2\right)^T &= -\left(\mathbf{v}_{A,i}^1\right)^T\mathbf{A}_2, i = 1,\ldots,p. \qquad (63)
\end{aligned}$$

Proof: We will first prove (a) and (b) together. To this end, let $(\lambda,\mathbf{u})$ be an eigenvalue and the corresponding right eigenvector of $\mathbf{A}$. Upon defining $\mathbf{u} := \left[\mathbf{u}_1^T\mathbf{u}_2^T\right]^T$ with $\mathbf{u}_1,\mathbf{u}_2 \in \mathbb{C}^{Jp\times 1}$, and recalling that $\mathbf{A}\mathbf{u} = \lambda\mathbf{u}$, or equivalently $\left(-\lambda^2\mathbf{I} + \lambda(\mathbf{I}-\mathbf{A}_1) - \mathbf{A}_2\right)\mathbf{u}_1 = \mathbf{0}$ we deduce that $\mathbf{u}_1^{\mathcal{H}}\left(-\lambda^2\mathbf{I} + \lambda(\mathbf{I}-\mathbf{A}_1) - \mathbf{A}_2\right)\mathbf{u}_1 = \mathbf{0}$, where $\mathcal{H}$ denotes conjugate transposition. The next step is to express λ as a function of $\mathbf{u}_1$, as well as the entries of $\mathbf{A}_1, \mathbf{A}_2$ and show that this function has amplitude less than one. Utilizing the structure of matrices $\mathbf{A}_1$, $\mathbf{A}_2$ in (36) and letting $\mathbf{u}_1 = \left[\mathbf{u}_{1,1}^T \ldots \mathbf{u}_{1,J}^T\right]^T$ with $\mathbf{u}_{1,j} \in \mathbb{C}^{p\times 1}$ we obtain

$$\begin{aligned}
&-\left(\sum_{j=1}^{J}\mathbf{u}_{1,j}^{\mathcal{H}}\mathbf{B}_j\mathbf{u}_{1,j}\right)\lambda^2 + \left[\sum_{j=1}^{J}\mathbf{u}_{1,j}^{\mathcal{H}}(\mathbf{B}_j - c_j|\mathcal{B}_j|\mathbf{I})\mathbf{u}_{1,j}\right.\\
&\left. + 2\sum_{b\in\mathcal{B}}\frac{\left\|\sum_{j'\in\mathcal{N}_b} c_{j'}\mathbf{u}_{1,j'}\right\|_2^2}{\sum_{\beta\in\mathcal{N}_b} c_\beta}\right]\lambda - \sum_{b\in\mathcal{B}}\frac{\left\|\sum_{j'\in\mathcal{N}_b} c_{j'}\mathbf{u}_{u,j'}\right\|_2^2}{\sum_{\beta\in\mathcal{N}_b} c_\beta} = 0.
\end{aligned}$$

The roots of the latter second order equation with respect to λ are $\lambda_{1,2} = \alpha_1 \pm \sqrt{\alpha_1^2 - \alpha_2}$, where

$$\begin{aligned}
\alpha_1 = \frac{1}{2}\left(\sum_{j=1}^{J}\mathbf{u}_{1,j}^{\mathcal{H}}\mathbf{B}_j\mathbf{u}_{1,j}\right)^{-1}&\left(\sum_{j=1}^{J}\mathbf{u}_{1,j}^{\mathcal{H}}(\mathbf{B}_j - c_j|\mathcal{B}_j|\mathbf{I})\mathbf{u}_{1,j}\right.\\
&\left.+2\sum_{b\in\mathcal{B}}\frac{\left\|\sum_{j'\in\mathcal{N}_b} c_{j'}\mathbf{u}_{1,j'}\right\|_2^2}{\sum_{\beta\in\mathcal{N}_b} c_\beta}\right) \qquad (64)
\end{aligned}$$

$$\alpha_2 = \left(\sum_{j=1}^{J}\mathbf{u}_{1,j}^{\mathcal{H}}\mathbf{B}_j\mathbf{u}_{1,j}\right)^{-1}\left(\sum_{b\in\mathcal{B}}\frac{\left\|\sum_{j'\in\mathcal{N}_b} c_{j'}\mathbf{u}_{1,j'}\right\|_2^2}{\sum_{\beta\in\mathcal{N}_b} c_\beta}\right). \qquad (65)$$

Based on (64) and (65) we will show that $|\lambda_{1,2}| < 1$. If $\alpha_1^2 < \alpha_2$, then $\lambda_{1,2}$ is complex with nonzero imaginary part, which implies that $|\lambda_{1,2}|^2 = \alpha_2$. Applying the Cauchy-Schwartz inequality on (65) we find:

$$\begin{aligned}
\sum_{b\in\mathcal{B}}\frac{\|\sum_{j'\in\mathcal{N}_b} c_{j'}\mathbf{u}_{1,j'}\|_2^2}{\sum_{\beta\in\mathcal{N}_b} c_\beta} &\leq \sum_{b\in\mathcal{B}}\sum_{j'\in\mathcal{N}_b} c_{j'}\|\mathbf{u}_{1,j'}\|_2^2\\
&= \sum_{j=1}^{J} c_j|\mathcal{B}_j|\mathbf{u}_{1,j}^{\mathcal{H}}\mathbf{u}_{1,j}. \qquad (66)
\end{aligned}$$

Using the definition of $\mathbf{B}_j$ we obtain $\sum_{j=1}^{J}\mathbf{u}_{1,j}^{\mathcal{H}}\mathbf{B}_j\mathbf{u}_{1,j} = 2\sum_{j=1}^{J}\mathbf{u}_{1,j}^{\mathcal{H}}\mathbf{H}_j^T\boldsymbol{\Sigma}_{n_jn_j}^{-1}\mathbf{H}_j\mathbf{u}_{1,j} + \sum_{j=1}^{J}c_j|\mathcal{B}_j|\|\mathbf{u}_{1,j}\|_2^2$. Since the first summand of the latter is strictly positive, we infer that $|\lambda_{1,2}|^2 = \alpha_2 < 1$. Now if $\alpha_1^2 \geq \alpha_2$ then $\lambda_{1,2}$ is real and using (66) we obtain that $\alpha_1 < 1$, from which we deduce that $\lambda_{1,2} = \alpha_1 - \sqrt{\alpha_1^2 - \alpha_2} < 1$ and $\lambda_{1,2} \geq 0$. The last case we consider is $\lambda_{1,2} = \alpha_1 + \sqrt{\alpha_1^2 - \alpha_2}$. Applying (66) to the numerator of α_1 it follows directly that

$$\begin{aligned}
\alpha_1 &\leq \frac{2\sum_{j=1}^{J}\mathbf{u}_{1,j}^{\mathcal{H}}\mathbf{H}_j^T\boldsymbol{\Sigma}_{x_jx_j}^{-1}\mathbf{H}_j\mathbf{u}_{1,j} + 2\sum_{j=1}^{J}c_j|\mathcal{B}_j|\|\mathbf{u}_{1,j}\|_2^2}{2\sum_{j=1}^{J}\mathbf{u}_{1,j}^{\mathcal{H}}\mathbf{B}_j\mathbf{u}_{1,j}}\\
&= 1 - \frac{\sum_{j=1}^{J}\mathbf{u}_{1,j}^{\mathcal{H}}\mathbf{H}_j^T\boldsymbol{\Sigma}_{x_jx_j}^{-1}\mathbf{H}_j\mathbf{u}_{1,j}}{\sum_{j=1}^{J}\mathbf{u}_{1,j}^{\mathcal{H}}\mathbf{B}_j\mathbf{u}_{1,j}}. \qquad (67)
\end{aligned}$$

$$\lambda_{1,2} \leq 1-\alpha_3+\sqrt{\alpha_3^2+\alpha_2\left(\frac{2\sum_{j=1}^{J}\mathbf{u}_{1,j}^{\mathcal{H}}\mathbf{H}_j^T\boldsymbol{\Sigma}_{x_jx_j}^{-1}\mathbf{H}_j\mathbf{u}_{1,j}+\sum_{b\in\mathcal{B}}(\sum_{\beta\in\mathcal{N}_b}c_\beta)^{-1}\|\sum_{j'\in\mathcal{N}_b}c_{j'}\mathbf{u}_{1,j'}\|_2^2}{\sum_{j=1}^{J}\mathbf{u}_{1,j}^{\mathcal{H}}\mathbf{B}_j\mathbf{u}_{1,j}}-1\right)}.$$

For brevity, let α_2 denote the second term in the rhs of (67). Using (67) only for the first summand in $\lambda_{1,2}$ and factorizing the expression $\alpha_1^2-\alpha_2$ inside the square root, we obtain the equation shown at the top of the page. Equation (66) implies that the maximum value the square root can attain is α_3; thus, $\lambda_{1,2} \leq 1$. Strict equality, holds when $\sum_{b\in\mathcal{B}}\|\sum_{j'\in\mathcal{N}_b}c_{j'}\mathbf{u}_{1,j'}\|_2^2\left(\sum_{\beta\in\mathcal{N}_b}c_\beta\right)^{-1} = \sum_{j=1}^{J}c_j|\mathcal{B}_j|\|\mathbf{u}_{1,j}\|_2^2$, or equivalently if and only if $\mathbf{u}_1 = \beta\left[\boldsymbol{\epsilon}_i^T\ldots\boldsymbol{\epsilon}_i^T\right]^T \in \mathbb{C}^{Jp\times 1}$ for $i=1,\ldots,p$. As β^{-1} can be absorbed in the corresponding left eigenvector, we set $\beta=1$ w.l.o.g. But since a right eigenvector $\mathbf{u}$ associated with the $\lambda=1$, satisfies $\mathbf{A}\mathbf{u}=\mathbf{u}$, we have that $\mathbf{u}_1=\mathbf{u}_2$. Thus, $\lambda=1$ if and only if $\mathbf{u}=\left[\boldsymbol{\epsilon}_i^T\ldots\boldsymbol{\epsilon}_i^T\right]^T\in\mathbb{C}^{2Jp\times 1}$, for $i=1,\ldots,p$. Furthermore, since $\mathbf{A}\left[\boldsymbol{\epsilon}_i^T\ldots\boldsymbol{\epsilon}_i^T\right]^T = \left[\boldsymbol{\epsilon}_i^T\ldots\boldsymbol{\epsilon}_i^T\right]^T$ for $i=1,\ldots,p$, the geometric multiplicity of $\lambda=1$ is p.

The remaining step is to show that the algebraic multiplicity of $\lambda=1$ is also p. Due to space limitations, we only sketch the proof of the latter which relies on the Jordan canonical form $\mathbf{A}=\mathbf{T}\mathbf{J}_A\mathbf{T}^{-1}$, where $\mathbf{T}\in\mathbb{C}^{2Jp\times 2Jp}$ is invertible and $\mathbf{J}_A\in\mathbb{C}^{2Jp\times 2Jp}$ is block diagonal matrix. Matrix $\mathbf{J}_A$ contains p diagonal blocks associated with the eigenvalue $\lambda=1$ whose structure can be found in [6]. Let $\mathbf{J}_{A,l_i}$ for $i\in[1,p]$ denote the ith of those diagonal blocks with size $l_i\times l_i$. Note that $\sum_{i=1}^p l_i$ equals the algebraic multiplicity of $\lambda=1$[6]. It suffices to have $\{l_i=1\}_{i=1}^p$, which we prove by contradiction. Specifically, we assume that $\exists\mathbf{J}_{A,i}$ for which $l_i\geq 2$, and try to solve the system of equations $\mathbf{A}\mathbf{T}=\mathbf{T}\mathbf{J}_A$ which turns out not to have a solution allowing us to conclude that $l_i=1$ for $i\in[1,p]$.

Next, we proceed with part (c). The p dominant left eigenvectors (corresponding to the eigenvalue 1), denoted by $\{\mathbf{v}_{A,i}\}_{i=1}^p$, satisfy $\mathbf{v}_{A,i}^T\mathbf{A}=\mathbf{v}_{A,i}^T$, through which we obtain the equivalent conditions:

$$\begin{aligned}\left(\mathbf{v}_{A,i}^1\right)^T\mathbf{A}_1 &= \left(\mathbf{v}_{A,i}^2\right)^T\\ \left(\mathbf{v}_{A,i}^1\right)^T\mathbf{A}_2+\left(\mathbf{v}_{A,i}^2\right)^T &= \mathbf{0}.\end{aligned}\tag{68}$$

Combining the two equations in (68) and using the fact that $\mathbf{v}_{A,i}^T\mathbf{u}_{A,i}=1$ for $i\in[1,2Jp]$ we arrive at

$$\begin{aligned}\left(\mathbf{v}_{A,i}^1\right)^T(\mathbf{I}_{Jp}-\mathbf{A}_1-\mathbf{A}_2) &= \left(\mathbf{v}_{A,i}^1\right)^T\\ (\mathbf{v}_{A,i}^1)^T(\mathbf{I}-\mathbf{A}_2)\mathbf{u}_{A,1,i} &= 1,\end{aligned}\tag{69}$$

where the second equation comes from (68), and $\mathbf{u}_{A,1,i}=\left[\boldsymbol{\epsilon}_i^T\ldots\boldsymbol{\epsilon}_i^T\right]^T$ contains the first Jp entries of the corresponding right eigenvector $\mathbf{u}_{A,i}$. Note that (69) provides sufficient conditions for $\mathbf{v}_{A,i}$ to be a left eigenvector which are satisfied by setting $\mathbf{v}_{A,i}^1$ as suggested in (c), while the vector $\mathbf{v}_{A,i}^2$ can be obtained from the second equation in (68). This concludes the proof of Lemma 6. □

To proceed with the proof of Proposition 4, we rely on the matrix eigendecomposition to write $\mathbf{A}^{n-1}=\sum_{i=1}^{2Jp}\lambda_{A,i}^{n-1}\mathbf{u}_{A,i}\mathbf{v}_{A,i}^T$. Using Lemma 6 (a), it follows directly that $\mathbf{A}_\infty:=\lim_{k\to\infty}\mathbf{A}^{k-1}=\lim_{k\to\infty}\mathbf{A}^k=\sum_{i=1}^p\mathbf{u}_{A,i}\mathbf{v}_{A,i}^T$. Since the noise terms in (41) have zero mean, we further have

$$\begin{aligned}\lim_{k\to\infty}E\left[\delta\bar{\boldsymbol{\phi}}(k+1)\right] &= \mathbf{A}_\infty\left[(\mathbf{I}-\mathbf{A}_1)^T\ \mathbf{I}\right]^T\hat{\mathbf{x}}-\mathbf{A}_\infty\\ &\quad\times\left[\mathbf{A}_1^T-\mathbf{I}\right]^T\boldsymbol{\phi}(0)-\mathbf{A}_\infty[\mathbf{A}_2^T\ \mathbf{I}]^T\boldsymbol{\phi}(-1)\end{aligned}\tag{70}$$

with $\boldsymbol{\phi}(0)=\hat{\mathbf{x}}$ and $\boldsymbol{\phi}(-1)=\mathbf{0}$. Next, we show that the second and third summand in (70) are zero. Indeed, using the first equation in (68) we obtain $\mathbf{A}_\infty\cdot\left[\mathbf{A}_1^T-\mathbf{I}\right]^T=\sum_{i=1}^p\mathbf{u}_{A,i}\left[\left(\mathbf{v}_{A,i}^1\right)^T\mathbf{A}_1-\left(\mathbf{v}_{A,i}^2\right)^T\right]=\mathbf{0}$, while through the second equation in (68) we have $\mathbf{A}_\infty[\mathbf{A}_2\ \ \mathbf{I}]^T=\sum_{i=1}^p\mathbf{u}_{A,i}\left[{\mathbf{v}_{A,i}^1}^T\mathbf{A}_2+{\mathbf{v}_{A,i}^2}^T\right]=\mathbf{0}$. It follows that

$$\begin{aligned}&\lim_{k\to\infty}E\left[\delta\bar{\boldsymbol{\phi}}(k+1)\right]\\ &=\mathbf{A}_\infty[(\mathbf{I}-\mathbf{A}_1)^T\ \mathbf{I}]^T\hat{\mathbf{x}}=\sum_{i=1}^p\mathbf{u}_{A,i}(\mathbf{v}_{A,i}^1)^T\hat{\mathbf{x}}\\ &=\sum_{i=1}^p\mathbf{u}_{A,i}\boldsymbol{\epsilon}_i^T\left[\left(\sum_{j=1}^J\mathbf{H}_j^T\boldsymbol{\Sigma}_{x_jx_j}^{-1}\mathbf{H}_j\right)^{-1}\sum_{j=1}^J\mathbf{H}_j^T\boldsymbol{\Sigma}_{x_jx_j}^{-1}\mathbf{x}_j\right]\\ &=(\mathbf{1}_J\otimes\mathbf{I}_p)\hat{\mathbf{s}}_{BL},\end{aligned}\tag{71}$$

where the vector $\mathbf{1}_J\in\mathbb{R}^{J\times 1}$ denotes the $J\times 1$ vector of all ones. The second equality in (71) follows from (68), and the third one using (63).

Now we proceed to find the limit of noise covariance matrix in (43) as $k\to\infty$. Toward this end, let $\boldsymbol{\Sigma}_A$ denote the matrix between $\mathbf{v}_{A,i}^T$ and $\mathbf{v}_{A,i'}$ in (45). Starting from (43) we can write

$$\begin{aligned}\boldsymbol{\Sigma}_\eta(k+1) &= \bar{\boldsymbol{\Sigma}}_{\eta\eta_b}+\sum_{n=0}^{k-1}\left(\sum_{i=1}^{2Jp}\lambda_{A,i}^n\mathbf{u}_{A,i}\mathbf{v}_{A,i}^T\right)\\ &\quad\times\boldsymbol{\Sigma}_A\left(\sum_{i'=1}^{2Jp}\lambda_{A,i'}^n\mathbf{v}_{A,i'}\mathbf{u}_{A,i'}^T\right)\\ &= \bar{\boldsymbol{\Sigma}}_{\eta\eta_b}+\sum_{i=p+1}^{2Jp}\sum_{i'=p+1}^{2Jp}\left(\sum_{n=0}^{k-1}(\lambda_{A,i}\lambda_{A,i'})^n\right)\\ &\quad\times\left(\mathbf{v}_{A,i}^T\boldsymbol{\Sigma}_A\mathbf{v}_{A,i'}\right)\mathbf{u}_{A,i}\mathbf{u}_{A,i'}^T\end{aligned}\tag{72}$$

where the second equality follows because $\mathbf{v}_{A,i}^T\boldsymbol{\Sigma}_A\mathbf{v}_{A,i'}=0\ \forall\ i,i'\in[1,p]$. Using the fact that $|\lambda_{A,i}\lambda_{A,i'}|<1$ for $i,i'\in[p+1,2Jp]$, we obtain $\sum_{n=0}^\infty(\lambda_{A,i}\lambda_{A,i'})^n=(1-\lambda_{A,i}\lambda_{A,i'})^{-1}$ and (45) follows. □

REFERENCES

[1] S. Barbarossa and G. Scutari, "Decentralized maximum likelihood estimation for sensor networks composed of nonlinearly coupled dynamical systems," *IEEE Trans. Signal Process.*, vol. 55, no. 7, pp. 3456–3470, Jul. 2007.

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

[3] D. Blatt and A. Hero, "Distributed maximum likelihood estimation for sensor networks," in *Proc. Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, Montreal, QC, Canada, May 2004, pp. 929–932.

[4] V. Delouille, R. Neelamani, and R. Baraniuk, "Robust distributed estimation in sensor networks using the embedded polygons algorithm," in *Proc. 3rd Int. Symp. Info. Processing Sensor Networks*, Berkeley, CA, Apr. 2004, pp. 405–413.

[5] A. Dogandžič and B. Zhang, "Distributed estimation and detection for sensor networks using hidden Markov random field models," *IEEE Trans. Signal Process.*, vol. 54, no. 8, pp. 3200–3215, Aug. 2006.

[6] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1999.

[7] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[8] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Control*, vol. 49, pp. 1520–1533, Sep. 2004.

[9] R. Olfati-Saber and J. S. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *Proc. 44th IEEE Conf. Dec. Eur. Control Conf.*, Seville, Spain, Dec. 2005, pp. 6698–6703.

[10] L. Prescosolido, S. Barbarossa, and G. Scutari, "Decentralized detection and localization through sensor networks designed as a population of self-synchronized oscillators," in *Proc. Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, Toulouse, France, May 2006, pp. 981–984.

[11] M. G. Rabbat, R. D. Nowak, and J. A. Bucklew, "Generalized consensus algorithms in networked systems with erasure links," in *Proc. 4th Int. Symp. Info. Processing Sensor Networks*, New York, Jun. 2005, pp. 1088–1092.

[12] A. Ribeiro and G. B. Giannakis, "Bandwidth-constrained distributed estimation for wireless sensor networks—Part II: Unknown probability density function," *IEEE Trans. Signal Process.*, vol. 54, no. 7, pp. 2784–2796, Jul. 2006.

[13] D. Scherber and H. C. Papadopoulos, "Distributed computation of averages over *ad hoc* networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 4, pp. 776–787, Apr. 2005.

[14] I. D. Schizas and G. B. Giannakis, "Consensus-based distributed estimation of random signals with wireless sensor networks," presented at the 40th Asilomar Conf. Signals, Systems, Computers, Monterey, CA, Oct. 2006.

[15] D. P. Spanos, R. Olfati-Saber, and R. J. Murray, "Distributed sensor fusion using dynamic consensus," presented at the 16th IFAC World Congr., Prague, Czech Republic, Jul. 2005.

[16] D. P. Spanos, R. Olfati-Saber, and R. M. Murray, "Dynamic consensus on mobile networks," presented at the 16th IFAC World Congr., Prague, Czech, Jul. 2005.

[17] J.-J. Xiao, A. Ribeiro, Z.-Q. Luo, and G. B. Giannakis, "Distributed compression-estimation using wireless sensor networks," *IEEE Signal Process. Mag.*, vol. 23, no. 4, pp. 27–41, Jul. 2006.

[18] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Syst. Control Lett.*, vol. 53, pp. 65–78, Sep. 2004.

[19] L. Xiao, S. Boyd, and S.-J. Kim, "Distributed average consensus with least-mean-square deviation," *J. Parallel Distrib. Comput.*, vol. 67, pp. 33–46, Jan. 2007.

[20] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. 4th Int. Symp. Inf. Processing Sensor Networks*, Berkeley, CA, Apr. 2005, pp. 63–70.

Ioannis D. Schizas received the Diploma in computer engineering and informatics with Honors from the University of Patras, Patras, Greece, in 2004 and the M.Sc. degree in electrical and computer engineering from the University of Minnesota, MN, in 2007.

Since August 2004, he has been working toward the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis. His research interests lie in the areas of communication theory, signal processing, and networking. His current research focuses on distributed signal processing with wireless sensor networks, and distributed compression and source coding.

Alejandro Ribeiro received the B.Sc. degree in electrical engineering from the Universidad de la Republica Oriental del Uruguay, Montevideo, in 1998. He received the M.Sc. and Ph.D. degrees in electrical engineering in May 2005 and December 2006, respectively, from the University of Minnesota, Minneapolis.

He is currently a Research Associate with the Department of Electrical and Computer Engineering, University of Minnesota. From 1998 to 2003, he was a member of the Technical Staff at Bellsouth Montevideo. His research interests lie in the areas of communication theory, signal processing, and networking. His current research focuses on wireless cooperative communications, random access, wireless *ad hoc* and sensor networks, and distributed signal processing.

Dr. Ribeiro is a Fulbright Scholar.

Georgios B. Giannakis (F'97) received the Diploma in electrical engineering from the National Technical University of Athens, Greece, 1981. He received the M.Sc. degree in electrical engineering in 1983, the M.Sc. degree in mathematics in 1986, and the Ph.D. degree in electrical engineering in 1986, all from the University of Southern California (USC), Los Angeles.

Since 1999, he has been a professor with the Department of Electrical and Computer Engineering, University of Minnesota, where he now holds an ADC Chair in Wireless Telecommunications. His general interests span the areas of communications, networking, and statistical signal processing—subjects on which he has published more than 250 journal papers, 450 conference papers, two edited books, and two upcoming research monographs on *Space-Time Coding for Broadband Wireless Communications* (New York: Wiley, 2006) and *Ultra-Wideband Wireless Communications* (Cambridge, U.K.: Cambridge Press, 2007). Current research focuses on diversity techniques, complex-field and space-time coding, multicarrier, cooperative wireless communications, cognitive radios, cross-layer designs, mobile *ad hoc* networks, and wireless sensor networks.

Dr. Giannakis is the (co-)recipient of six paper awards from the IEEE Signal Processing (SP) and Communications Societies including the G. Marconi Prize Paper Award in Wireless Communications. He also received Technical Achievement Awards from the SP Society (2000), from EURASIP (2005), a Young Faculty Teaching Award, and the G. W. Taylor Award for Distinguished Research from the University of Minnesota. He has served the IEEE in a number of posts.