Distributed In-Network Channel Decoding

Hao Zhu, *Student Member, IEEE*, Georgios B. Giannakis, *Fellow, IEEE*, and Alfonso Cano, *Student Member, IEEE*

***Abstract*—Average log-likelihood ratios (LLRs) constitute sufficient statistics for centralized maximum-likelihood block decoding as well as for *a posteriori* probability evaluation which enables bit-wise (possibly iterative) decoding. By acquiring such average LLRs per sensor it becomes possible to perform these decoding tasks in a low-complexity *distributed* fashion using wireless sensor networks. At affordable communication overhead, the resultant distributed decoders rely on local message exchanges among single-hop neighboring sensors to achieve iteratively consensus on the average LLRs per sensor. Furthermore, the decoders exhibit robustness to non-ideal inter-sensor links affected by additive noise and random link failures. Pairwise error probability bounds benchmark the decoding performance as a function of the number of consensus iterations. Interestingly, simulated tests corroborating the analytical findings demonstrate that only a few consensus iterations suffice for the novel distributed decoders to approach the performance of their centralized counterparts.**

***Index Terms*—Channel coding, decoding, distributed detection, wireless sensor networks (WSNs).**

I. Introduction

COLLABORATION among wireless sensors capitalizes on the premise that limited individual processing and communication capabilities can be greatly enhanced when sharing information with neighboring sensors. This basic idea has been exploited in various areas including distributed detection, classification and estimation, to name a few [10], [15], [17]. The present paper focuses on how resource-constrained wireless sensors can collaboratively decode messages broadcasted by an access point (AP). Due to conceivably low signal-to-noise-ratio (SNR) conditions, each low-cost sensor may be unable to reliably decode the broadcasted message. However, by exchanging just minimal sufficient information with its single-hop neighbors, each sensor can potentially decode the AP message with accuracy as high as if it had received all other sensors' data. This holds promise to bring local probability of error performance as close as possible to global (i.e., centralized) performance.

Decoding amounts to deciding among multiple candidate codewords and, in this sense, the distributed decoders pursued here are related to the distributed detectors developed using wireless sensor networks (WSNs) in; e.g., [10] and [17]. However, these detectors are not suitable for channel decoding problems because their communication complexity grows with the number of hypotheses, which is exponential in the message size.

Compared with fusion center (FC) based operation, distributed *in-network* processing is based on sensor-to-sensor message exchanges and offers two attractive features: i) it scales favorably with regards to power requirements, thus prolonging the WSN lifetime; and ii) it avoids isolating the point(s) of failure, thus enhancing fault tolerance and WSN survivability. Indeed, remote sensors added to expand the geographical area covered by a large-scale WSN have to consume increased power to reach the FC(s); and if the FC fails, the WSN falls in outage. By contrast, if sensors implement in-network processing, information is exchanged only among one-hop neighbors, which keeps the communication overhead per sensor at an affordable level within its neighborhood, as the WSN size grows. Furthermore, error rate performance degrades gracefully even when inter-sensor communications fail, so long as the network remains connected. As a result, it is more likely that the WSN can survive for a longer time.

The present paper considers two specific problems of interest: a) distributed in-network maximum-likelihood (ML) block decoding; and b) distributed in-network *a posteriori* probability (APP) evaluation. The ultimate goal is to obtain tools such as distributed versions of the Viterbi algorithm for block ML decoding and the Bahl–Cocke–Jelinek–Raviv (BCJR) algorithm for APP evaluation to enable (possibly iterative) decoding [12, Sec. 8.8]. The solution of a) and b) becomes possible by having sensors consent on minimal sufficient statistics, namely average log-likelihood ratios (LLRs) of the distributed data. Compared to distributed detection approaches, consenting on the average LLR incurs communication complexity that grows linearly in the message length.

Achieving consensus across agents has been considered in vehicle coordination [7], as well as in distributed sample-averaging of sensor observations [8], [9], [21]. A more general consensus approach to handle distributed estimation problems (not necessarily expressible in terms of averages) can be found in [18] using the method of multipliers (MoM) [3, Sec. 3.4.4]. Relative to these contributions, the present work leverages existing consensus averaging (CA) approaches for distributed channel decoding in the presence of imperfect inter-sensor links. Specifically, two iterative algorithms are derived: i) CA with a single iteration (CA-SI); and ii) CA using the MoM (CA-MoM). The

Manuscript received December 27, 2008; accepted April 14, 2009. First published May 27, 2009; current version published September 16, 2009. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Walid Hachem. This work was supported by NSF Grants CCF 0830480 and CON 014658; and also through collaborative participation in the Communications and Networks Consortium sponsored by the U.S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon. This work was presented in part at the Conference on Information Sciences and Systems, Princeton University, Princeton, NJ, March 19–21, 2008.

The authors are with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455 USA (e-mail: zhuh@umn.edu; georgios@umn.edu; alfonso@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.2009.2023936

CA-SI algorithm offers resilience to inter-sensor link failures; yet under noisy links the variance of the resultant iterates grows unbounded. Introducing a vanishing stepsize, the CA-SI algorithm can ensure bounded variance at the price of slower convergence rate. On the other hand, the CA-MoM algorithm converges in the presence of inter-sensor noise without a vanishing stepsize. Compared to [18], the CA-MoM algorithm here offers two distinct novelties: it does not require a bridge sensor set with which sensors need to communicate, thus offering a *fully* distributed CA approach; and it is provably convergent in the presence of inter-sensor link failures.

Error performance bounds on the average bit-error rate (BER) as a function of the number of iterations are also derived. These bounds enable sensors to quantify the number of iterations needed to reach a prescribed error performance. Numerical tests demonstrate that only a few consensus iterations suffice for the novel distributed decoders to approach the performance of their centralized counterpart.

The rest of the paper is organized as follows. In Section II, the system model is introduced and the distributed ML block decoding and APP evaluation problems are formulated. Section III identifies sufficient statistics for these problems in terms of the average LLR. Section IV presents and compares the CA-SI and CA-MoM algorithms to estimate the average LLR in a distributed fashion under pragmatic assumptions on the inter-sensor communication links. Section V derives BER bounds for CA-based distributed decoding, and Section VI corroborates the analytical findings through simulated tests.

Notation: Upper (lower) bold face letters will be used for matrices (column vectors); $[\cdot]_{ij}$ ($[\cdot]_i$) for the (i,j)th (i th) entry of a matrix (vector); $\mathrm{diag}(\mathbf{x})$ is a diagonal matrix with $\mathbf{x}$ on its diagonal; $(\cdot)^T$ denotes transposition; $(\cdot)^{\mathcal{H}}$ Hermitian transpose; $\mathbf{I}_N$ the $N \times N$ identity matrix; $\mathbf{1}_N$ the $N \times 1$ vector of all ones; $\mathbf{0}_N$ the $N \times 1$ vector of all zeros; $\|\cdot\|$ the Frobenius norm; $|\cdot|$ the cardinality of a set; $(\mathcal{C})\mathcal{N}(\mu, \sigma^2)$ a (complex) Gaussian distribution with mean μ and variance σ^2; $p[x = z|y]$ the probability density function (pdf) of the continuous random variable (r.v.) x, given the r.v. y, evaluated at z; $p[x = z|y]$ the probability mass function (pmf) of the discrete r.v. x, given the r.v. y, evaluated at z.

II. Modeling and Problem Statement

Consider the access point (AP) depicted in Fig. 1 broadcasting an $N \times 1$ coded block $\mathbf{x}$ to a set of J wireless low-cost, power-limited sensors. The WSN is modeled as a graph $\mathcal{G} := \{\mathcal{E}, \mathcal{J}\}$, where $\mathcal{J} := \{1, \ldots, J\}$ denotes the set of sensors, and $\mathcal{E} \subset \mathcal{J} \times \mathcal{J}$ the set of available communication links (graph edges). The set of neighbors of a given sensor j is denoted by $\mathcal{N}_j \subseteq \mathcal{J}$. Graph connectivity is captured by the $J \times J$ adjacency matrix $\mathbf{A}$, where $[\mathbf{A}]_{ji} = 1$ if $i \in \mathcal{N}_j$, otherwise $[\mathbf{A}]_{ji} = 0$. The graph Laplacian matrix is defined as $\mathbf{L} := \mathbf{D} - \mathbf{A}$, where $\mathbf{D} := \mathrm{diag}(|\mathcal{N}_1|, \ldots, |\mathcal{N}_J|)$ is the degree matrix.

Let $\mathbf{y}_j$ denote the $N \times 1$ block of symbols received at sensor j. The system operates under the following assumptions:

(a1) The WSN is connected, i.e., there is a (possibly multi-hop) path connecting any two nodes;

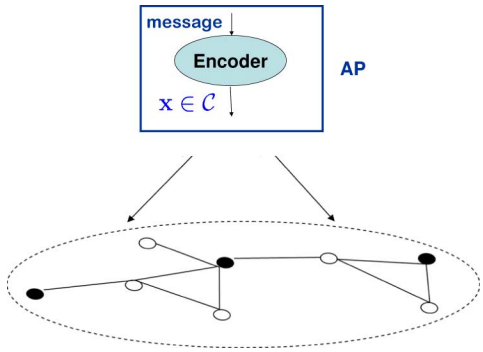


Fig. 1. System model.

(a2) Sensors communicate with single-hop neighbors over symmetric links that can be:
  (a2.1) ideal and time-invariant per-block; or
  (a2.2) time-invariant corrupted by additive zero-mean noise with variance σ_i^2 at sensor $i \in \mathcal{N}_j$, and uncorrelated across j and i; or
  (a2.3) time-varying in the sense that the link between neighboring sensors j and i is allowed to fail with probability $1 - p_{ji}$, where $0 < p_{ji} < 1$.

(a3) Entry x_n of $\mathbf{x}$ is binary $\{0, 1\}$ and all sensors know the codebook $\mathcal{C}$ that $\mathbf{x}$ belongs to;

(a4) AP-sensor channels are discrete, memoryless, symmetric, and conditionally independent across sensors; and sensor j knows the conditional pdf $p[\mathbf{y}_j|\mathbf{x}]$ characterizing its channel with the AP.

Note that the connectivity in (a1) need not be strong, meaning that a sensor must have just enough power to reach single-hop neighbors but not all other sensors in the network. The ideal (virtually error-free) inter-sensor communications under (a2.1) are possible with sufficiently powerful error control codes; and are considered to benchmark the imperfect link performance allowed under (a2.2) and (a2.3). Similar to [18], the noisy inter-sensor link model (a2.2) includes quantization errors and receiver noise. The random failures considered in (a2.3) encompass i) communication among sensors using error-detection codes such as cyclic redundancy check (CRC) codes and ii) fading channels. Link failures under (a2.3) are different from sensor failures. The former allow for link recovery since $p_{ij} > 0$, whereas the latter are associated with permanent link failures. Binary codewords are assumed in (a3) for simplicity in exposition, but generalizations to any finite alphabet are possible as shown in Section III. Channels with finite memory are also covered by (a4) provided that e.g., multi-carrier modulation is employed [12, Ch. 11.2.3]. Channel knowledge in (a4) is assumed acquired via training. The AP-sensor link models include the binary symmetric channel, case in which the probability in (a4) depends only on the cross-over probability when $x_n \in \{0, 1\}$; and the binary phase shift keying (BPSK) transmission over the additive white Gaussian noise (AWGN) channel, case in which $\mathbf{y}_j = \sqrt{E_s}(2\mathbf{x} - \mathbf{1}_N) + \boldsymbol{\epsilon}_j$, $j \in \mathcal{J}$, where E_s denotes average symbol energy and $\boldsymbol{\epsilon}_j$ the AWGN.

Given $\mathbf{y}_j$ per sensor j, the objective is to find under a1)–a4) and without a fusion center *distributed* solutions of the following:

- the *centralized maximum-likelihood (ML)* block decoder

$$(\text{P1}) \quad \hat{\mathbf{x}}_{ML} = \arg\max_{\mathbf{x}\in\mathcal{C}} p[\mathbf{y}_1,\ldots,\mathbf{y}_J|\mathbf{x}]; \tag{1}$$

- the *centralized a posteriori probability (APP)* evaluator

$$(\text{P2}) \quad \Pr[x_n|\mathbf{y}_1,\ldots,\mathbf{y}_J] \quad \forall n=1,\ldots,N. \tag{2}$$

The ultimate goal is to obtain tools such as *distributed* versions of the Viterbi algorithm (DiVA) for ML decoding and the BCJR algorithm (DiBCJR) for APP evaluation to enable (possibly iterative) decoding via in-network message exchanges among sensors. It is also desirable to have such algorithms incur low computational cost and affordable overhead in inter-sensor communications. Furthermore, the design must be geared toward efficient operation in the power-limited regime that is typical of WSN applications; i.e., maximizing data rates is not of prime interest here.

Remark 1 : (*Decoding as hypothesis testing*) Due to the conditional independence in (a4), ML decoding of equiprobable codewords amounts to a multiple hypotheses test aiming to maximize the sum of per-sensor log-likelihoods, i.e., $\hat{\mathbf{x}}_{ML} = \arg\max_{\mathbf{x}\in\mathcal{C}} \sum_{j=1}^{J} \log p(\mathbf{y}_j|\mathbf{x})$. Taking linear (N,K) binary block codes as an example, the number of hypotheses equals the cardinality of the codebook $|\mathcal{C}| = 2^K$; or, generally $|\mathcal{C}| = Q^K$ for a finite alphabet of size Q. Viewed as a test of multiple hypotheses, distributed ML decoding with centralized performance is possible so long as each sensor can acquire the average log-likelihood over all sensors *for each candidate* $\mathbf{x} \in \mathcal{C}$. To obtain this average, belief consensus and belief propagation approaches were developed in [10] and [17] in the context of distributed detection. Belief consensus in [10] relies on consensus averaging (CA)–a scheme originally developed in [9], [21] for distributed estimation of sample averages. Relative to the aforementioned contributions, the present paper exploits the special structure of the channel decoding problem and aims at low-communication-complexity solutions with quantifiable error performance and pragmatic inter-sensor communication links.

III. Consensus-Based Distributed Decoding

In this section, (P1) and (P2) are cast in equivalent forms amenable to distributed implementation. Since AP-sensor channels are memoryless and independent across sensors [c.f. (a4)], the log-likelihood of $\{\mathbf{y}_j\}_{j=1}^J$ given $\mathbf{x}$ is

$$\begin{aligned}\log p[\mathbf{y}_1,\ldots,\mathbf{y}_J|\mathbf{x}] &= \sum_{j=1}^{J} \log p[\mathbf{y}_j|\mathbf{x}] \\ &= \sum_{j=1}^{J}\sum_{n=1}^{N} \log p[y_{jn}|x_n]. \end{aligned} \tag{3}$$

Furthermore, with $x_n \in \{0,1\}$, each summand in (3) can be written as

$$\log p[y_{jn}|x_n] = -x_n \log\left(\frac{p[y_{jn}|x_n=0]}{p[y_{jn}|x_n=1]}\right) + \log p[y_{jn}|x_n=0] \tag{4}$$

where clearly the term $\log p[y_{jn}|x_n=0]$ is constant with respect to (w.r.t.) the variable x_n. Upon defining the local log-likelihood ratio (LLR) corresponding to the nth bit of the codeword at sensor j as

$$\gamma_{jn} := \log\left(\frac{p[y_{jn}|x_n=0]}{p[y_{jn}|x_n=1]}\right) \tag{5}$$

the centralized ML decoding problem in (1) is equivalent to

$$\hat{\mathbf{x}}_{ML} = \arg\min_{\mathbf{x}\in\mathcal{C}} \sum_{j=1}^{J}\sum_{n=1}^{N} \gamma_{jn}x_n = \arg\min_{\mathbf{x}\in\mathcal{C}} \bar{\boldsymbol{\gamma}}^T\mathbf{x} \tag{6}$$

where

$$\bar{\boldsymbol{\gamma}} := \frac{1}{J}\sum_{j=1}^{J} \boldsymbol{\gamma}_j \tag{7}$$

and the vector $\boldsymbol{\gamma}_j := [\gamma_{j1},\ldots,\gamma_{jN}]^T$ collects the N scalar local LLRs per sensor j.

Equation (6) reveals that ML optimal decoding amounts to a linear program [5]. But since $\mathcal{C}$ is available to all sensors [c.f. (a3)], to solve (6) in a distributed fashion it suffices for each sensor to acquire the $N\times 1$ average LLR vector $\bar{\boldsymbol{\gamma}}$. Correspondingly, Appendix A shows that for a general alphabet of size Q each sensor must consent to $(Q-1)$ such average LLR vectors (one per pair of alphabet values). It is worth stressing though that, unlike [10], [17], the communication complexity here does not depend on the codebook size $|\mathcal{C}| = Q^K$. The next example provides further insight on this point.

Example 1 (Sufficient consensus with AWGN AP-sensor channels): Consider the AWGN model $\mathbf{y}_j = \sqrt{E_s}(2\mathbf{x}-\mathbf{1}_N) + \boldsymbol{\epsilon}_j$, where $\boldsymbol{\epsilon}_j \sim \mathcal{N}(0, N_{0,j}\mathbf{I}_N/2)$. The ML decoding problem in this special case amounts to

$$\begin{aligned}\hat{\mathbf{x}}_{ML} = \arg\max_{\mathbf{x}\in\mathcal{C}} &\left(\sum_{j=1}^{J} \frac{2\sqrt{E_s}\mathbf{y}_j}{JN_{0,j}}\right)^T (2\mathbf{x}-\mathbf{1}_N) \\ &- \left(\sum_{j=1}^{J}\frac{E_s}{JN_{0,j}}\right)\|2\mathbf{x}-\mathbf{1}_N\|^2. \end{aligned} \tag{8}$$

Since $(2x_n-1)$ takes values $\{-1,1\}$, the term $\|2\mathbf{x}-\mathbf{1}_N\|^2$ in (8) is constant and thus irrelevant to the maximization. Upon comparing (8) with (6), the average LLR here is $\bar{\boldsymbol{\gamma}} = -1/J\sum_{j=1}^{J} 4\sqrt{E_s}\mathbf{y}_j/N_{0,j}$. This is reasonable since the weighted average data in the AWGN model constitute sufficient statistics for ML decoding [12, Ch. 5]. Notice that the local receive SNR is $2E_s/N_{0,j}$, whereas the global receive SNR after maximum-ratio combining is $\sum_{j=1}^{J} 2E_s/N_{0,j}$, which is precisely the SNR of $\bar{\gamma}_n$, $\forall n$. Intuitively, if sensors acquire in a distributed fashion their weighted average data, they can subsequently carry out ML decoding with their local SNR boosted to its global value—thanks to averaging—that is otherwise reachable only when all data $\{\mathbf{y}_j\}_{j=1}^J$ are centrally available.

Let us now turn our attention from ML block decoding to the APP evaluation task in (P2). In this problem, sensor $j \in \mathcal{J}$ knows $p[\mathbf{y}_j|\mathbf{x}]$ and seeks $\Pr[x_n|\mathbf{y}_1,\ldots,\mathbf{y}_J]$ for $n=1,\ldots,N$.

The latter is completely determined by the following ratio (since numerator and denominator must sum up to one):

$$l_n := \frac{\Pr[x_n=0|\mathbf{y}_1,\ldots,\mathbf{y}_J]}{\Pr[x_n=1|\mathbf{y}_1,\ldots,\mathbf{y}_J]} = \frac{\sum_{\mathbf{x}\in\mathcal{C}_n^0} p[\mathbf{y}_1,\ldots,\mathbf{y}_J|\mathbf{x}]}{\sum_{\mathbf{x}\in\mathcal{C}_n^1} p[\mathbf{y}_1,\ldots,\mathbf{y}_J|\mathbf{x}]} \quad (9)$$

$$= \frac{\sum_{\mathbf{x}\in\mathcal{C}_n^0} \exp\left[-\left(\sum_{j=1}^{J}\boldsymbol{\gamma}_j\right)^T\mathbf{x}\right]}{\sum_{\mathbf{x}\in\mathcal{C}_n^1} \exp\left[-\left(\sum_{j=1}^{J}\boldsymbol{\gamma}_j\right)^T\mathbf{x}\right]} = \frac{\sum_{\mathbf{x}\in\mathcal{C}_n^0}\exp\left[-J\bar{\boldsymbol{\gamma}}^T\mathbf{x}\right]}{\sum_{\mathbf{x}\in\mathcal{C}_n^1}\exp\left[-J\bar{\boldsymbol{\gamma}}^T\mathbf{x}\right]} \quad (10)$$

where $\mathcal{C}_n^i := \{\mathbf{x}\in\mathcal{C}|x_n = i\}$ for $i\in\{0,1\}$ is the set of codewords in $\mathcal{C}$ whose nth entry is equal to i; and (9) is due to Bayes' theorem while (10) follows from (3)–(5). Again, (10) reveals that if each sensor can acquire the average LLR vector $\bar{\boldsymbol{\gamma}}$ in a distributed fashion, distributed APP evaluation is also possible regardless of the codebook size $|\mathcal{C}| = Q^K$. Summarizing, it has been established that:

Proposition 1: *For distributed ML decoding and APP evaluation tasks to attain performance of their centralized counterparts under (a1)–(a4), it suffices for each sensor to obtain, i.e., consent on the average LLR vector $\bar{\boldsymbol{\gamma}}$—a task incurring linear communication complexity in the codeword size N.*

The average LLR is the sufficient statistic for a number of centralized decoding algorithms [16, Ch. 4]. The major implication of Proposition 1 is that acquiring the average LLR in a distributed fashion gives rise to distributed versions of these algorithms: DiVA, DiBCJR, and distributed iterative (LDPC or turbo) decoding algorithms. The key enabler of these schemes is distributed evaluation of the average LLR, the topic dealt with in the ensuing section.

IV. Consensus on Average LLR

The major issue in Proposition 1 is whether the average LLR vector in (7) can indeed be found distributedly even when sensors can only communicate with one-hop neighbors. This will be shown possible through iterative consensus schemes tailored for the setup described by (a1)–(a4). The resulting algorithms will generate (local) iterates $\bar{\gamma}_{jn}(k)$, which can be viewed as *local estimates* of the nth entry $\bar{\gamma}_n$ of $\bar{\boldsymbol{\gamma}}$ in (7) per sensor j and iteration k. These local estimates will be required to converge to $\bar{\gamma}_n$. But more importantly, if the iterations stop before finding the exact $\bar{\gamma}_n$, each sensor will be allowed to decode using the estimate available up to that instant. Therefore, an approximate sufficient statistic for (P1) and (P2) will become available per sensor j and iteration k.

If the network operates under (a2.1) or (a2.3), the average $\bar{\boldsymbol{\gamma}}$ in (7) can be found in a distributed fashion by applying the CA algorithm of [9], [13], [21], which is based on a single iterative (SI) equation involving $\bar{\gamma}_{jn}(k)$. However, when inter-sensor links suffer from noise as in (a2.2), these algorithms fail to converge. The solution advocated in [6] enables convergence at the price of considerably reducing convergence speed. Motivated by these facts, this paper will introduce novel CA alternatives based on the method of multipliers (CA-MoM), in which the average value is obtained as the iterative solution of a convex optimization problem. Modified iterations of the CA-MoM will be further developed to cope with random link failures in (a2.3).

A. CA-SI Algorithms

With its local information $\{\gamma_{jn}\}_{n=1}^N$ available, sensor j seeks the global $\bar{\boldsymbol{\gamma}}$ in (7); i.e., $\bar{\gamma}_n = 1/J\sum_{j=1}^J\gamma_{jn}$ for $n=1,\ldots,N$. The CA-SI algorithm yields iterates $\bar{\gamma}_{jn}(k)$ per sensor j converging to the desired $\bar{\gamma}_n\ \forall n$, as $k\to\infty$. In the present context, these iterates per sensor j exploit LLR information from single-hop neighboring sensors $\mathcal{N}_j$ through the following linear recursion:

$$\bar{\gamma}_{jn}(k) = \bar{\gamma}_{jn}(k-1) + \sum_{i\in\mathcal{N}_j} W_{ji}[\bar{\gamma}_{in}(k-1) - \bar{\gamma}_{jn}(k-1)], \quad j\in\mathcal{J},\ n=1,\ldots,N. \quad (11)$$

With the initialization $\bar{\gamma}_{jn}(0) = \gamma_{jn}$ and properly selected weights W_{ji}, convergence to $\bar{\gamma}_n\ \forall j\in\mathcal{J}$ as $k\to\infty$ is ensured [21]. With $\bar{\boldsymbol{\gamma}}_n(k) := [\bar{\gamma}_{1n}(k),\ldots,\bar{\gamma}_{Jn}(k)]^T$ and $W_{ji} = 0$ for $i\notin\mathcal{N}_j$, concatenating (11) for $j=1,\ldots,J$, yields the vector recursion $\bar{\boldsymbol{\gamma}}_n(k) = \mathbf{W}\bar{\boldsymbol{\gamma}}_n(k-1)$, where $[\mathbf{W}]_{ji\in\mathcal{J}} = W_{ji}$. The weight matrix $\mathbf{W}$ guarantees convergence if [21]: i) $\mathbf{1}^T\mathbf{W} = \mathbf{1}^T$; ii) $\mathbf{W}\mathbf{1} = \mathbf{1}$; and iii) $\rho(\mathbf{W} - (1/J)\mathbf{1}\mathbf{1}^T) < 1$, where $\rho(\cdot)$ denotes the spectral radius.

The iterations in (11) are known to exhibit resilience to random link failures [13]. However, if additive noise is present when neighboring sensors exchange $\bar{\gamma}_{jn}(k)$, the variance of $\bar{\gamma}_{jn}(k)$ in (11) grows unbounded with k [6]. To avoid the latter, [6] advocated scaling the correction term in (11) by a vanishing stepsize $\alpha(k)$; see also [2], [11], [14]. In this case, (11) is replaced by

$$\bar{\gamma}_{jn}(k) = \bar{\gamma}_{jn}(k-1) + \alpha(k)\sum_{i\in\mathcal{N}_j} W_{ji}[\bar{\gamma}_{in}(k-1) - (\bar{\gamma}_{jn}(k-1) + \eta_{jin}(k))] \quad (12)$$

where $\eta_{jin}(k)$ denotes additive noise at sensor j, and $\alpha(k)$ is chosen to satisfy

$$\alpha(k) > 0, \quad \sum_{k=1}^{\infty}\alpha(k) = \infty, \quad \text{and} \sum_{k=1}^{\infty}\alpha^2(k) < \infty. \quad (13)$$

The third condition in (13) dictates $\alpha(k)$ to vanish as $k\to\infty$, while the second condition demands this convergence to be sufficiently slow. Typical choice of a stepsize satisfying (13) is $\alpha(k) = 1/k$. All the iterates $\{\bar{\gamma}_{jn}(k)\}_{j=1}^J$ in (12) reach consensus to an estimator of $\bar{\gamma}_n$ that has bounded variance [6]; see also [8]. However, similar to stochastic approximation iterations encountered in adaptive signal processing, the price paid is considerably slower convergence and inability to track changes; see e.g., [19, Ch. 9] and also the simulations in Section VI. This renders the CA-SI less desirable for distributed decoding in the presence of noise. These considerations motivate our distributed, iterative CA decoder based on the method of multipliers.

B. CA-MoM Algorithm With Ideal Links

The CA-MoM approach originates from the well-known fact that the sample average can be viewed as the solution of a least-squares cost, namely $\bar{\gamma}_n = \arg\min_\theta 1/2 \sum_{j=1}^J (\gamma_{jn} - \theta)^2$. The crux of distributing this centralized cost is to use *local auxiliary* variables $\bar{\gamma}_{jn}$ for the wanted average per sensor j along with constraints to ensure agreement of these variables among neighboring sensors; hence, the term consensus. The variables $\bar{\gamma}_{jn}$ can be written as the minimizer of the following consensus-constrained quadratic cost function:

$$\begin{aligned} \min_{\bar{\boldsymbol{\gamma}}_n} \quad & \frac{1}{2}\sum_{j=1}^{J}(\gamma_{jn}-\bar{\gamma}_{jn})^2 \\ \text{s.t.} \quad & \bar{\gamma}_{jn}-\bar{\gamma}_{in}=0, \quad j\in\mathcal{J},\ i\in\mathcal{N}_j \end{aligned} \tag{14}$$

where $\bar{\boldsymbol{\gamma}}_n := [\bar{\gamma}_{1n}, \ldots, \bar{\gamma}_{Jn}]^T$. Since the network is connected under (a1), the neighborhood consensus constraint in (14) ensures that consensus is achieved over the entire network; i.e., $\bar{\gamma}_{jn} = \bar{\gamma}_{in}\ \forall j, i \in \mathcal{J}$. This provides a sufficient condition to guarantee that the optimum of (14) is achieved at the sample average.

The goal is to solve (14) distributedly using the MoM [3, Sec. 3.4.4]. The latter entails an additional set of variables $\boldsymbol{z}_n := \{\{z_{jin}, z'_{jin}\}_{i\in\mathcal{N}_j}\}_{j\in\mathcal{J}}$, through which (14) can be equivalently written as

$$\begin{aligned} \min_{\bar{\gamma}_j, \boldsymbol{z}_n} \quad & \sum_{j=1}^{J}\frac{1}{2}(\gamma_{jn}-\bar{\gamma}_{jn})^2 \\ \text{s.t.} \quad & \bar{\gamma}_{jn}-z_{jin}=0, \quad j\in\mathcal{J},\ i\in\mathcal{N}_j \\ & -\bar{\gamma}_{in}-z'_{jin}=0, \quad j\in\mathcal{J},\ i\in\mathcal{N}_j \\ & z_{jin}+z'_{jin}=0, \quad j\in\mathcal{J},\ i\in\mathcal{N}_j. \end{aligned} \tag{15}$$

Let v_{jin} and v'_{jin} denote the Lagrange multipliers associated with the constraints $\bar{\gamma}_{jn} - z_{jin} = 0$ and $-\bar{\gamma}_{in} - z'_{jin} = 0$, respectively. Likewise, define the set $\mathcal{C}_z := \{\boldsymbol{z}_n : z_{jin} + z'_{jin} = 0,\ \forall j \in \mathcal{J},\ i \in \mathcal{N}_j\}$ that represents the constraints on the entries of $\boldsymbol{z}_n$. With $c > 0$ denoting a penalty coefficient, consider the augmented Lagrangian function of (15)

$$\begin{aligned} &\mathcal{L}_a(\bar{\boldsymbol{\gamma}}_n, \boldsymbol{z}_n, \mathbf{v}_n, \mathbf{v}'_n) \\ &\quad = \frac{1}{2}\sum_{j=1}^{J}(\gamma_{jn}-\bar{\gamma}_{jn})^2 + \sum_{j=1}^{J}\sum_{i\in\mathcal{N}_j}\Big\{v_{jin}(\bar{\gamma}_{jn}-z_{jin}) \\ &\qquad\qquad + v'_{jin}(-\bar{\gamma}_{in}-z'_{jin})\Big\} \\ &\qquad + \frac{c}{2}\sum_{j=1}^{J}\sum_{i\in\mathcal{N}_j}\Big\{(\bar{\gamma}_{jn}-z_{jin})^2 + (-\bar{\gamma}_{in}-z'_{jin})^2\Big\} \end{aligned} \tag{16}$$

where the $2|\mathcal{E}| \times 1$ vector $\mathbf{v}_n := [\{v_{1in}\}_{i\in\mathcal{N}_1}, \ldots, \{v_{Jin}\}_{i\in\mathcal{N}_J}]^T$, and likewise for $\mathbf{v}'_n$.

The alternating-direction MoM operates by minimizing $\mathcal{L}_a$ in (16) cyclically w.r.t. one set of variables given the other variables. The $(k+1)$st iteration of the MoM solver of (16) becomes (see also [3, p. 255])

$$\bar{\boldsymbol{\gamma}}_n(k+1) = \arg\min_{\bar{\boldsymbol{\gamma}}_n} \mathcal{L}_a\left(\bar{\boldsymbol{\gamma}}_n, \boldsymbol{z}_n(k), \mathbf{v}_n(k), \mathbf{v}'_n(k)\right) \tag{17a}$$

$$\boldsymbol{z}_n(k+1) = \arg\min_{\boldsymbol{z}_n\in\mathcal{C}_z} \mathcal{L}_a\left(\bar{\boldsymbol{\gamma}}_n(k+1), \boldsymbol{z}_n, \mathbf{v}_n(k), \mathbf{v}'_n(k)\right) \tag{17b}$$

$$v_{jin}(k+1) = v_{jin}(k) + c(\bar{\gamma}_{jn}(k+1) - z_{jin}(k+1)), \quad j\in\mathcal{J},\ i\in\mathcal{N}_j \tag{17c}$$

$$v'_{jin}(k+1) = v'_{jin}(k) + c(-\bar{\gamma}_{in}(k+1) - z'_{jin}(k+1)), \quad j\in\mathcal{J},\ i\in\mathcal{N}_j \tag{17d}$$

where (17c) and (17d) are due to the fact that the multipliers in (16) appear linearly in $\mathcal{L}_a$.

It is proved in Appendix B that if $v_{jin}(k)$ and $v'_{jin}(k)$ are initialized identically $\forall (j,i) \in \mathcal{E}$; i.e., $v_{jin}(0) = v'_{jin}(0)$, the variables $\mathbf{v}'_n(k)$ and $\boldsymbol{z}_n(k+1)$ can be eliminated and (17a)–(17d) can be simplified to

$$v_{jin}(k) = v_{jin}(k-1) + \frac{c}{2}(\bar{\gamma}_{jn}(k) - \bar{\gamma}_{in}(k)), \quad j\in\mathcal{J},\ i\in\mathcal{N}_j \tag{18a}$$

$$\bar{\gamma}_{jn}(k+1) = \frac{1}{1+2c|\mathcal{N}_j|}\Big\{\gamma_{jn} - \sum_{i\in\mathcal{N}_j}\Big[v_{jin}(k) - v_{ijn}(k) - c(\bar{\gamma}_{jn}(k) + \bar{\gamma}_{in}(k))\Big]\Big\}, \ j\in\mathcal{J}. \tag{18b}$$

The iterations (18a)–(18b) constitute the CA-MoM algorithm. Sensor $j \in \mathcal{J}$ maintains the local estimate of the average LLR $\bar{\gamma}_{jn}(k)$ and all the multipliers $\{v_{jin}(k)\}_{i\in\mathcal{N}_j}$. During the kth iteration, sensor j receives the broadcasted estimates $\bar{\gamma}_{in}(k)$ from all its neighboring sensors $i \in \mathcal{N}_j$, and updates the corresponding multipliers via (18a). It then transmits back the updated multiplier $v_{jin}(k)$ to each of its neighboring sensors $i \in \mathcal{N}_j$, based on which each sensor j is able to determine $\bar{\gamma}_{jn}(k+1)$ via (18b). Subsequently, all sensors $j \in \mathcal{J}$ broadcast their updated estimates $\bar{\gamma}_{jn}(k+1)$ to their neighbors, thus completing the kth iteration and initializing the next one.

The iterates in (18a)–(18b) are provably convergent, as asserted in the following proposition.

Proposition 2: *(CA-MoM with ideal inter-sensor links) Under (a2.1), the CA-MoM iterations (18a)–(18b) with arbitrary initialization of $\bar{\gamma}_{jn}(1)$ and $v_{jin}(0)$, $\forall (j,i) \in \mathcal{E}$ and $c > 0$, reach consensus to the average LLR in (7) as $k \to \infty$; i.e.,*

$$\lim_{k\to\infty} \bar{\gamma}_{jn}(k) = \bar{\gamma}_n, \quad \forall j\in\mathcal{J},\ n = 1, \ldots, N. \tag{19}$$

Proof: Iterations (17a)–(17d) follow directly from the MoM approach in [3, p. 255]; and as shown in Appendix B, they are equivalent to (18a)–(18b). Since the cost function in (15) is convex and its constraints comply with [3, Assumption 4.1, p. 255], the iterates in (17a)–(17d) converge as established by [3, Prop. 4.2, p. 256]. The aforementioned equivalence readily establishes the convergence of (18a)–(18b) as well. ■

Remark 2: (*Comparison with [18]*) A MoM-based consensus approach was also developed in [18] for distributed estimation. However, to form the MoM cost in [18], a subset of sensors called bridge sensors is required. In turn, an algorithm to find the bridge sensor set is necessary, and has to be re-run whenever a bridge sensor fails. Setting all sensors in the network to act as bridge sensors is possible, but the communication overhead is unnecessarily increased. Compared to

[18], the consensus-constrained formulation in (15) does not require such a bridge sensor set, and in this sense, it offers a *fully* distributed approach. In addition to bypassing the need for bridge sensors, the approach here relative to [18], has provable convergence when links fail randomly—an issue dealt with in Section IV-D.

Remark 3: (*Comparison with CA-SI for ideal links*) It has been shown that the CA-SI scheme in (11) can be derived using a primal-dual solver of (14) [13]. However, weight selections in [13] and [21] ensuring convergence of (11) without requiring knowledge of the global WSN topology per sensor, are rather limited. One possible choice is to select $\mathbf{W} = \mathbf{I}_J - (1/J)\mathbf{L}$, where $\mathbf{L}$ denotes the Laplacian of the graph. Since MoM-based iterations on the other hand are convergent $\forall\ c > 0$, they are more flexible to accelerate convergence speed, even when inter-sensor links are ideal; see also [22, p. 69]. In addition, MoM-based solvers can cope with noisy links as elaborated next.

C. CA-MoM Algorithm With Noisy Links

When inter-sensor links are corrupted by additive noise as in (a2.2), different from the CA-SI iterates in (11), the CA-MoM iterates in (18a)–(18b) will be shown to exhibit finite variance without resorting to a vanishing stepsize. To prove this claim, let $\eta_{jin}(k)$ and $\bar{\eta}_{jin}(k)$ denote the additive noise present in sensor $j \in \mathcal{J}$ when receiving $v_{ijn}(k)$ and $\bar{\gamma}_{in}(k)$ from sensor $i \in \mathcal{N}_j$, respectively. The iterations in (18a)–(18b) now take the form

$$v_{jin}(k) = v_{jin}(k-1) + \frac{c}{2}[\bar{\gamma}_{jn}(k) - (\bar{\gamma}_{in}(k) + \bar{\eta}_{jin}(k))], \quad j \in \mathcal{J},\ i \in \mathcal{N}_j \tag{20a}$$

$$\bar{\gamma}_{jn}(k+1) = \frac{1}{1+2c|\mathcal{N}_j|}\Big\{\gamma_{jn} - \sum_{i\in\mathcal{N}_j}[v_{jin}(k) - (v_{ijn}(k) + \eta_{jin}(k)) - c(\bar{\gamma}_{jn}(k) + \bar{\gamma}_{in}(k) + \bar{\eta}_{jin}(k))]\Big\}. \tag{20b}$$

Iterations (20a)–(20b) can be interpreted as stochastic gradient updates; see e.g., [3, Sec. 7.8]. Viewed from this vantage point, the noise causes $\bar{\gamma}_{jn}(k)$ to fluctuate around the noise-free optimal solution of (14). The magnitude of fluctuations is proportional to the noise variance. Our convergence claim in this case can be summarized as follows.

Proposition 3: (*CA-MoM with noisy inter-sensor links*) *Under (a2.2), the CA-MoM iterations (20a)–(20b) reach consensus to the average LLR in (7) in the mean sense as $k \longrightarrow \infty$; i.e.,*

$$\lim_{k\longrightarrow\infty} E\{\bar{\gamma}_{jn}(k)\} = \bar{\gamma}_n \quad \forall j \in \mathcal{J},\ n = 1,\ldots,N. \tag{21}$$

In addition, the variance of $\bar{\gamma}_{jn}(k)$ converges to a bounded value, which depends on the noise variance and the WSN topology.

Proof: See Appendix C. ∎

Remark 4: (*Comparison with CA-SI with noisy links*) Different from the CA-SI algorithm in (11), Proposition 3 ensures that as iterations evolve, the variance of $\bar{\gamma}_{jn}(k)$ remains bounded without a vanishing stepsize required by (12). As the simulations of Section VI will confirm, bypassing the need for a vanishing stepsize further enables the CA-MoM to converge faster than the noise-resilient CA-SI in (12).

D. CA-MoM Algorithm With Link Failures

In the presence of random link failures, the adjacency matrix $\mathbf{A}(k)$ at the kth iteration adheres to an independent and identically distributed (i.i.d.) Bernoulli process with $\Pr[[\mathbf{A}(k)]_{ji} = 1] = p_{ij} > 0$ as per (a2.3). Whenever a link fails between sensors, the corresponding multipliers become unavailable. The idea to cope with this challenge is to introduce an extra variable $v_{ijn}^{(j)}(k)$ per sensor j to keep track of neighbor i's multiplier information. The CA-MoM iterations (18a)–(18b) take now the form

$$v_{jin}(k) = \begin{cases} v_{jin}(k-1) + \frac{c}{2}[\bar{\gamma}_{jn}(k) - \bar{\gamma}_{in}(k)] & \text{if } [\mathbf{A}(k)]_{ji} = 1 \\ v_{jin}(k-1) & \text{if } [\mathbf{A}(k)]_{ji} = 0 \end{cases} \tag{22a}$$

$$v_{ijn}^{(j)}(k) = \begin{cases} v_{ijn}(k) & \text{if } [\mathbf{A}(k)]_{ji} = 1 \\ v_{ijn}^{(j)}(k-1) & \text{if } [\mathbf{A}(k)]_{ji} = 0 \end{cases} \tag{22b}$$

$$\bar{\gamma}_{jn}(k+1) = \frac{1}{1+2c|\mathcal{N}_j|}\Big\{\gamma_{jn} - \sum_{i\in\mathcal{N}_j} p_{ji}[v_{jin}(k) - v_{ijn}^{(j)}(k)] + 2c|\mathcal{N}_j|\bar{\gamma}_{jn}(k) + c\sum_{i\in\mathcal{N}_j(k)}(\bar{\gamma}_{in}(k) - \bar{\gamma}_{jn}(k))\Big\}. \tag{22c}$$

Sensor j locally stores the multipliers $\{v_{ijn}^{(j)}(k)\}_{i\in\mathcal{N}_j}$ of its neighboring sensors and freezes the multiplier pair $\{v_{jin}(k), v_{ijn}^{(j)}(k)\}$ whenever the link (j,i) fails at the kth iteration. Notice that p_{ji} is assumed known when updating $\bar{\gamma}_{jn}(k)$ in (22c). This way, sensor j keeps track of the unavailable multipliers by updating the local ones $v_{ijn}^{(j)}(k)$ through (22b), since the freezing strategy maintains the multiplier pair unchanged during the link failures.

Since each sensor j does not have to wait until information $\{\bar{\gamma}_{in}(k)\}_{i\in\mathcal{N}_j}$ from all its neighbors becomes available in order to update $\bar{\gamma}_{jn}(k+1)$ in (22c), iterations (22a)–(22c) can be seen as an *asynchronous* implementation of the CA-MoM. Convergence of the algorithm can be asserted in the mean-square sense (m.s.s.) as summarized next.

Proposition 4: (*CA-MoM with random failures of inter-sensor links*) *Under (a2.3), the iterations (22a)–(22c) consent on the average LLR of (7) in the m.s.s. as $k \longrightarrow \infty$; i.e.,*

$$\lim_{k\longrightarrow\infty} E\left\{|\bar{\gamma}_{jn}(k) - \bar{\gamma}_n|^2\right\} = 0, \quad \forall j \in \mathcal{J},\ n = 1,\ldots,N. \tag{23}$$

Proof: See Appendix D. ∎

Compared to the CA-MoM with ideal links, the asynchronous implementation here only requires extra information on the link probability p_{ji} to update the $\bar{\gamma}_{jn}(k+1)$ in (22c). It is possible to relax (a2.3) and allow for a real-time acquisition of p_{ji}. Specifically, at iteration k sensor j can substitute the running estimate $p_{ji}(k) = 1/k\sum_{k'=1}^{k}[\mathbf{A}(k')]_{ji}$ for p_{ji} into (22c). With the i.i.d. Bernoulli process in (a2.3), the asymptotic convergence of $p_{ji}(k)$ to the true p_{ji} is guaranteed, and (23) holds even when p_{ji} is acquired on-the-fly.

V. Performance Analysis

In Section IV, suitable iterative algorithms to solve (P1) and (P2) were developed under conditions (a2.1)–(a2.3) regarding inter-sensor communications. This section derives bounds for the BER performance of the CA-SI and CA-MoM algorithms as a function of the number of iterations k, even if consensus on the *exact* LLR vector has not been achieved. Recall also that ML block decoding minimizes the block error rate, whereas bitwise decoding using the APP minimizes the BER. For this reason, the analysis here will be focused on finding bounds for the BER of distributed ML decoders, with the understanding that those also bound the bitwise distributed APP decoder performance as well.

A. Pairwise Probability Bound

The probability of error P_e of the ML decoder in (1) can be bounded in terms of the pairwise error probability (PEP), which is the probability of erroneously decoding $\mathbf{x}$ as $\mathbf{x}' \in \mathcal{C}$ with $\mathbf{x}' \neq \mathbf{x}$. Letting $P_{\mathbf{x} \to \mathbf{x}'}$ denote the PEP, it holds that $P_e \leq \sum_{\mathbf{x} \in \mathcal{C}} P_{\mathbf{x}} \sum_{\mathbf{x}' \neq \mathbf{x}} P_{\mathbf{x} \to \mathbf{x}'}$, where $P_{\mathbf{x}}$ is the probability of transmitting $\mathbf{x}$. For a given LLR vector $\boldsymbol{\gamma}$, the PEP can be expressed as [c.f. (6)]

$$\begin{aligned} P_{\mathbf{x} \to \mathbf{x}'} &= \Pr\left[\boldsymbol{\gamma}^T(\mathbf{x}-\mathbf{x}') \geq 0 | \mathbf{x}\right] \\ &\leq E\left\{\exp\left[\lambda \boldsymbol{\gamma}^T(\mathbf{x}-\mathbf{x}')\right] | \mathbf{x}\right\} \end{aligned} \tag{24}$$

where the inequality follows from the Chernoff bound, see, e.g., [12, Sec. 2.4], and is satisfied for all $\lambda > 0$. Upon setting $\boldsymbol{\gamma} = \bar{\boldsymbol{\gamma}}$, the PEP in (24) corresponds to that of the centralized decoder, whereas if $\boldsymbol{\gamma} = \boldsymbol{\gamma}_j$, the PEP pertains to the local performance at sensor j.

Because the AP-sensor channels are memoryless, the Chernoff bound in (24) reduces to

$$P_{\mathbf{x} \to \mathbf{x}'} \leq \prod_{n=1}^{N} \left[\int_{\gamma_n} \exp\left[\lambda \gamma_n (x_n - x_n')\right] p[\gamma_n | x_n] d\gamma_n \right]. \tag{25}$$

Notice that $(x_n - x_n')$ can only take values 1, 0 or -1. Moreover, under (a4), the LLR is symmetrically distributed; hence, $p[\gamma_n | x_n = 0] = p[-\gamma_n | x_n = 1]$. The upper bound in (25) can thus be rewritten as

$$P_{\mathbf{x} \to \mathbf{x}'} \leq \prod_{n=1}^{N} \left(\Delta^{(\lambda)}\right)^{|x_n - x_n'|} \tag{26}$$

where, due to symmetry, it suffices to consider only the case $x_n = 0$ in

$$\Delta^{(\lambda)} := \int_{\gamma_n} \exp(-\lambda \gamma_n) p[\gamma_n | x_n = 0] d\gamma_n. \tag{27}$$

Parameter $\Delta^{(\lambda)}$ encapsulates the channel effects in decoding performance. It is independent of the codebook and the bit index n since the channel statistics are assumed invariant w.r.t. n.

To make the upper bound in (26) as tight as possible, consider the minimum achievable $\Delta := \min_{\lambda > 0} \Delta^{(\lambda)}$. Not surprisingly, Δ plays the same role as the Chernoff or Bhatacharyya parameters in the error performance analysis in [12, Sec. 6.8]. Here, Δ is derived as a function of the sufficient statistic $\boldsymbol{\gamma}$, whereas in [12, Sec. 6.8] it is expressed in terms of the received block $\mathbf{y}$. Clearly, for AWGN channels $\mathbf{y}$ and $\boldsymbol{\gamma}$ are closely related, as illustrated in the following example.

Example 2 (Performance analysis for AWGN AP-sensor channels): Consider again the AWGN AP-sensor channel using BPSK modulation as in Example 1. The input-output relationship per bit can be written as

$$y_{jn} = \sqrt{E_s}(2x_n - 1) + \epsilon_{jn}, \quad x_n = 0, 1 \tag{28}$$

where $\epsilon_{jn} \sim \mathcal{N}(0, N_{0,j}/2)$. The conditional probability is

$$p[y_{jn} | x_n] = \frac{1}{\sqrt{\pi N_{0,j}}} \exp\left\{ \frac{-[y_{jn} - \sqrt{E_s}(2x_n - 1)]^2}{N_{0,j}} \right\} \tag{29}$$

and the LLR can be explicitly expressed as

$$\gamma_{jn} = -\frac{4\sqrt{E_s} y_{jn}}{N_{0,j}} = \frac{4E_s}{N_{0,j}}(1 - 2x_n) - \frac{4\sqrt{E_s}\epsilon_{jn}}{N_{0,j}}. \tag{30}$$

Thus, the LLR γ_{jn} of the AP-sensor channel is a Gaussian random variable with conditional distribution $p(\gamma_n | x_n = 0) = \mathcal{N}(\mu_j, \sigma_j^2)$, where $\mu_j := 4E_s/N_{0,j}$ and $\sigma_j^2 := 8E_s/N_{0,j}$. Substituting the latter into (27), and minimizing w.r.t. λ leads to

$$\Delta = \exp\left(-\frac{E_s}{N_{0,j}}\right) \tag{31}$$

which coincides with the Bhatacharyya parameter for AWGN channels in [12, Sec. 6.8].

The subsequent steps to derive the BER bound are standard: once the parameter Δ is determined, the error performance can be upper bounded by a polynomial in Δ with coefficients that depend on the specific codeword structure, and are independent of the underlying channel [12, Sec. 6.8].

The next objective is to find similar bounds for the error performance of the local decoder using $\bar{\boldsymbol{\gamma}}_j(k)$, the average LLR estimate available per iteration k.

B. Error Performance Per Iteration

The key to evaluating BER performance per iteration k is to specify the distribution of $\bar{\boldsymbol{\gamma}}_j(k)$ as a function of k. To encompass both CA-SI and CA-MoM algorithms in the subsequent analysis, the following lemma is needed.

Lemma 1: *Under (a2.1), the LLR iterates in (11), or (18a)–(18b) can be commonly expressed as*

$$\bar{\gamma}_{jn}(k) = \sum_{i=1}^{J} c_{ji}(k) \gamma_{in}, \quad \forall j \in \mathcal{J},\ n = 1, \ldots, N \tag{32}$$

where the coefficients $c_{ji}(k)$ depend solely on the network topology.

Proof: See Appendix E. ∎

The implication of Lemma 1 is that $\{\bar{\boldsymbol{\gamma}}_j(k)\}_{j \in \mathcal{J}}$ maintain the memoryless and symmetry properties from the downlink channel [c.f. (a2)]. As a result, $\Delta^{(\lambda)}(k)$ in (27) becomes

$$\Delta_j^{(\lambda)}(k) := E\left\{ e^{-\lambda \bar{\gamma}_{jn}(k)} | x_n = 0 \right\} = \prod_{i=1}^{J} \Delta_i^{(\lambda / c_{ji}(k))} \tag{33}$$

where $\Delta_j^{(\lambda)}$ corresponds to the Δ parameter for the local channel associated with $\boldsymbol{\gamma}_j$.

Lemma 1 asserts a linear relationship for ideal inter-sensor links. Generalization to the noisy links in (a2.2) is possible given that this will only cause an additive equivalent noise term in (32) (c.f. Appendix C). Hence, under (a2.2) (33) can be modified to include the noise effects accordingly.

To obtain a tight upper bound for the BER at sensor j, let $\Delta_j(k) := \min_{\lambda>0} \Delta_j^{(\lambda)}(k)$ represent the equivalent channel at iteration k and sensor j. The parameter $\Delta_j(k)$ cannot be generally obtained in closed form for an arbitrary pdf $p[\gamma_n|x_n]$. It is however possible for AWGN AP-sensor channels, as demonstrated in the next example.

Example 3 (PEP bound for AWGN AP-sensor channels): Consider again BPSK transmission over AWGN AP-sensor channels as in Examples 1 and 2. Similar to Example 2, the local LLR for $x_n = 0$ is Gaussian with mean μ_j and variance σ_j^2. Under (a2.1), the average LLR estimate $\bar{\gamma}_{jn}(k)$ in (32) is a linear combination of Gaussian random variables. Hence, it is Gaussian distributed, with mean and variance given, respectively, by

$$\bar{\mu}_j(k) = \sum_{i=1}^{J} c_{ji}(k)\mu_i, \quad \text{and} \quad \bar{\sigma}_j^2(k) = \sum_{i=1}^{J} c_{ji}^2(k)\sigma_i^2. \quad (34)$$

If zero-mean inter-sensor noise is present as in (a2.2), the average LLR estimate $\bar{\gamma}_{jn}(k)$, in either (12) or (20a)–(20b), remains Gaussian distributed, with mean and variance given respectively by

$$\bar{\mu}_j(k) = \sum_{i=1}^{J} c_{ji}(k)\mu_i, \text{ and } \bar{\sigma}_j^2(k) = \sum_{i=1}^{J} c_{ji}^2(k)\sigma_i^2 + \Sigma_j(k) \quad (35)$$

where per iteration k, $\Sigma_j(k)$ is the concatenated equivalent noise variance at sensor j and can be obtained from the noise level σ_i^2 at sensor $i \in \mathcal{N}_j$. For example, Appendix C provides its closed-form expression for the CA-MoM algorithm, where $\Sigma_j(k) = [\boldsymbol{\Sigma}_{\bar{\gamma}_n}(k)]_{jj}$ with $\boldsymbol{\Sigma}_{\bar{\gamma}_n}(k)$ given in (61). Similar argument also holds for the CA-SI algorithm.

Under both (a2.1) and (a2.2) and given the respective mean and variance, (33) now becomes [c.f. (31)]

$$\Delta_j(k) = \exp\left(-\frac{\bar{\mu}_j^2(k)}{2\bar{\sigma}_j^2(k)}\right). \quad (36)$$

As a closing note, it is worth mentioning that unlike most existing applications of consensus to distributed detection and estimation problems, which assert estimation/detection performance only asymptotically (as $k \to \infty$), the present section evaluates error performance for a finite number of iterations. This is particularly important for channel decoding because it allows one to quantify the number of iterations needed to attain a prescribed BER value.

VI. Simulations

The CA-SI and CA-MoM distributed solvers of (P1) and (P2) are tested and compared in this section under various channel settings. The WSN includes $J = 10$ sensors uniformly distributed over the unit square $[0,1] \times [0,1]$. The communication range of each sensor is $r = 0.5$. Two nodes are connected if their Euclidean distance is less than r. As a result, the number of graph edges here is $|\mathcal{E}| = 18$. The channel between AP and sensor j is modeled as AWGN with SNR $2E_s/N_{0,j}$. For simplicity, the SNR is set to be the same at all sensors; i.e., $N_{0,j} = N_0 \, \forall j$. Matrix $\mathbf{W}$ of the CA-SI in Section IV-A is given by $\mathbf{W} = \mathbf{I} - \xi\mathbf{L}$, where $\xi = 1/(\lambda_2 + \lambda_J)$, and λ_i is the ith eigenvalue of $\mathbf{L}$, as in [21].

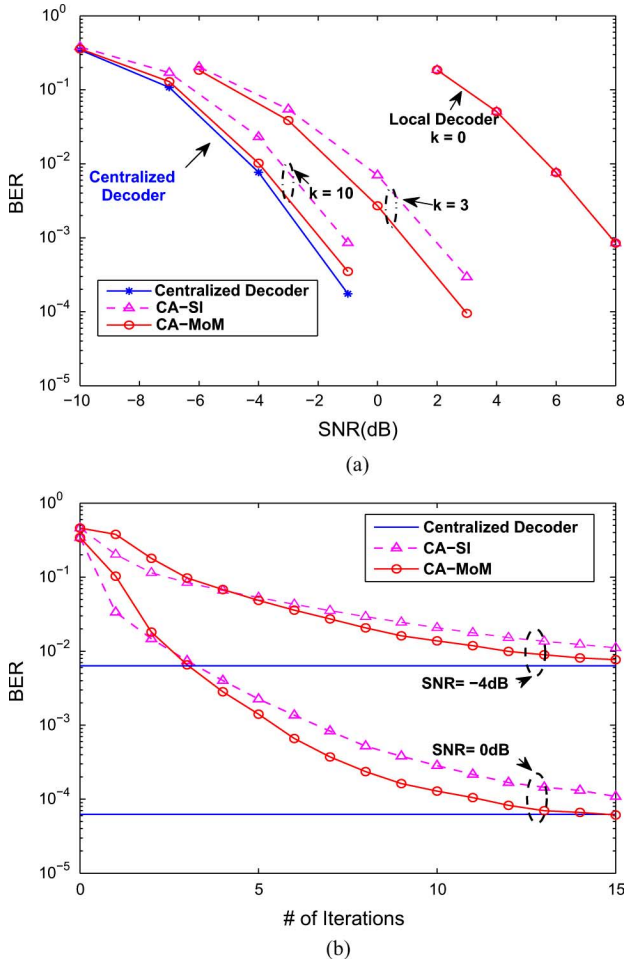


Fig. 2. Distributed ML decoder performance using convolutional codes ($N = 60$, rate 2/3) with ideal inter-sensor links for the centralized, local, and distributed decoders: (a) BER versus SNR (in dB) curves with variable number of consensus iterations; (b) BER versus number of consensus iterations for different AP-sensor SNRs.

Test Case 1: (*Block decoding with ideal inter-sensor links*) In this case, the average BER of the block ML decoder is simulated for ideal inter-sensor communication links [c.f. (a2.1)]. The AP encodes messages of $K = 40$ bits using a rate $2/3$ convolutional code $(N = 60)$ with generator registers $[4, 3]$ in octal form. Upon reception, each sensor initiates iterations with its neighbors to estimate the average LLR in (7). The iterations in (11) are used for the CA-SI algorithm, whereas the iterations in (18a)–(18b) are used for the CA-MoM algorithm. After a given number of iterations k, each sensor relies on the average LLR $\bar{\boldsymbol{\gamma}}(k)$ collected up to that instant to run the DiVA decoder. Fig. 2(a) compares the resulting average BER as a function of

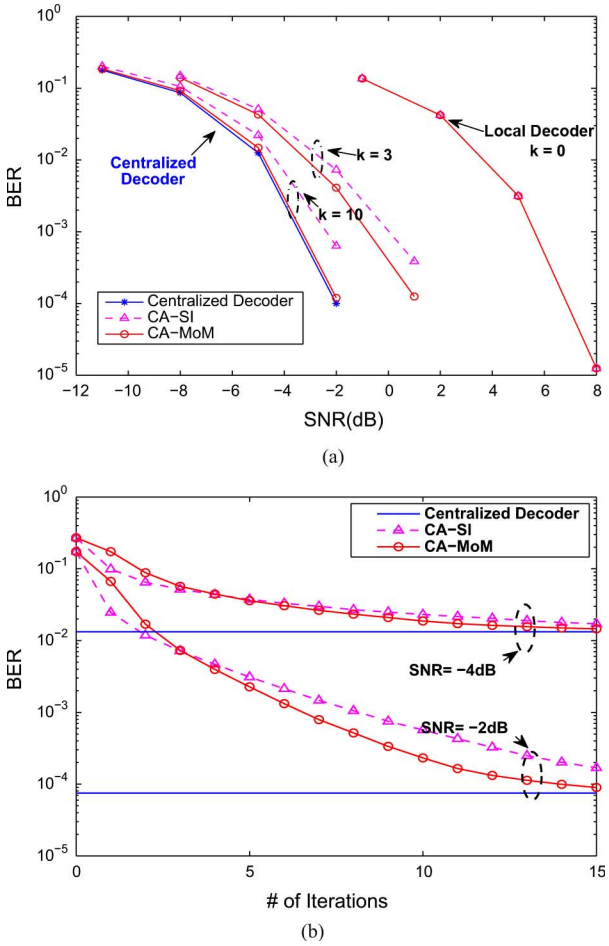


Fig. 3. Distributed MAP decoder performance using (7,4) Hamming codes with ideal inter-sensor links for the centralized, local, and distributed decoders: (a) BER versus SNR (in dB) curves with variable number of consensus iterations; (b) BER versus number of consensus iterations for different AP-sensor SNRs.

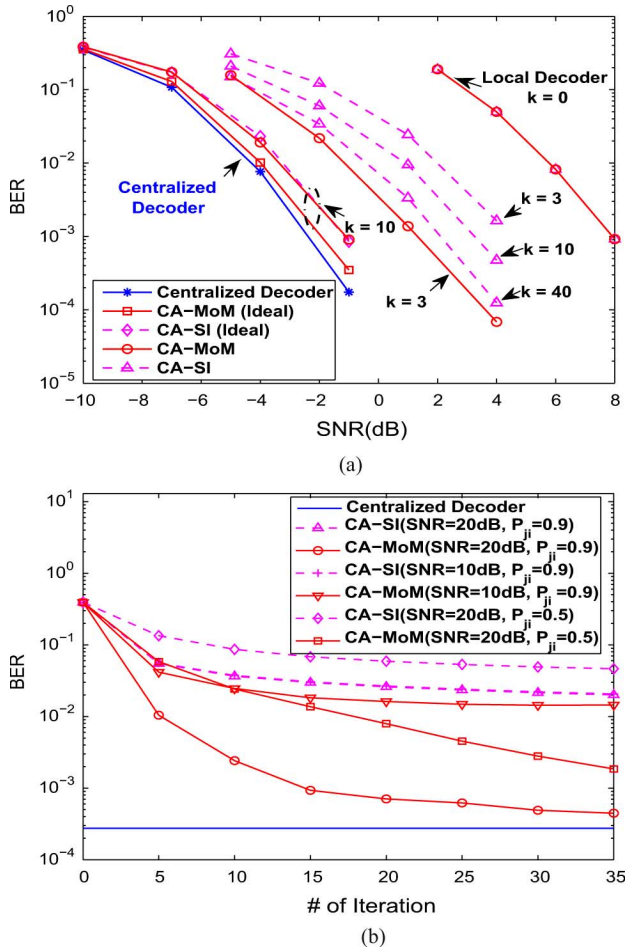


Fig. 4. Distributed ML decoder performance using convolutional codes ($N = 60$, rate 2/3) with imperfect inter-sensor links: (a) BER versus SNR (in dB) curves for various decoders with variable number of consensus iterations at $\mathrm{SNR} = 20$ dB (per sensor) and link failure probability $1 - p_{ji} = 0.1$; (b) BER versus number of consensus iterations with various inter-sensor link SNRs and failure probabilities at AP-sensor $\mathrm{SNR} = -1$ dB.

the SNR for different values of the iteration index k. The *centralized* decoder [using $\bar{\boldsymbol{\gamma}}$ in (7)] is included as a benchmark. The *local* decoding performance per sensor (corresponding to the distributed decoder after iteration number $k = 0$) is also included for comparison. At each iteration k there are JN transmissions throughout the network when running the CA-SI algorithm. For the CA-MoM algorithm this number is $(J + 2|\mathcal{E}|)N$. Clearly, the average BER sharply reduces as the number of iterations increases, approaching that of the centralized benchmark after a few iterations of both CA-SI and CA-MoM algorithms. This is not surprising since the diversity collected by the decoder increases as the LLR information from neighboring sensors iteratively becomes available. As seen, even for a few iterations, the CA-MoM algorithm outperforms the CA-SI, as predicted by Remark 3. Fig. 2(b) depicts the decoding performance as a function of the number of consensus iterations for different AP-sensor SNRs. The centralized decoder is shown as a benchmark for the distributed ones. It can be seen that the average BER sharply reduces after a few iterations.

Test Case 2: (*Bitwise decoding with ideal inter-sensor links*) In this test case, the AP encodes messages of $K = 4$ bits using the Hamming $(7, 4)$ code. As in Test Case 1, each sensor initiates iterations with its neighbors to estimate the average LLR $\bar{\boldsymbol{\gamma}}(k)$ and uses it to find the APP as in (10), which it employs subsequently to carry out the bitwise MAP decoder. Both Fig. 3(a) and (b) demonstrate similar relative performance of the two CA-SI and CA-MoM algorithms as in Test Case 1.

Test Case 3: (*Block decoding with imperfect inter-sensor links*) Fig. 4(a) depicts the average BER in the presence of both noise and random failures of inter-sensor links. The AP encodes messages as in the Test Case 1 and sensors decode using the VA. The additive noise power is 20 dB above the signal power for all links, whereas the probability of failure is set to $1 - p_{ji} = 0.1$ for all links. In the simulations, $\alpha(k)$ in (12) is set to be $1/k$. Fig. 4(a) compares the resulting average BER as a function of the SNR for different values of the iteration index k. The distributed decoders after $k = 10$ iterations with ideal AP-sensor links from Test Case 1 are included for comparison. Both centralized and local decoders are also included. The resulting BER of both algorithms degrades w.r.t. the one in Test Case 1. This is expected, since the additive noise in the inter-sensor communication links increases the uncertainty in the distributed ML decoding problem. Note, however, that the gap between the BER

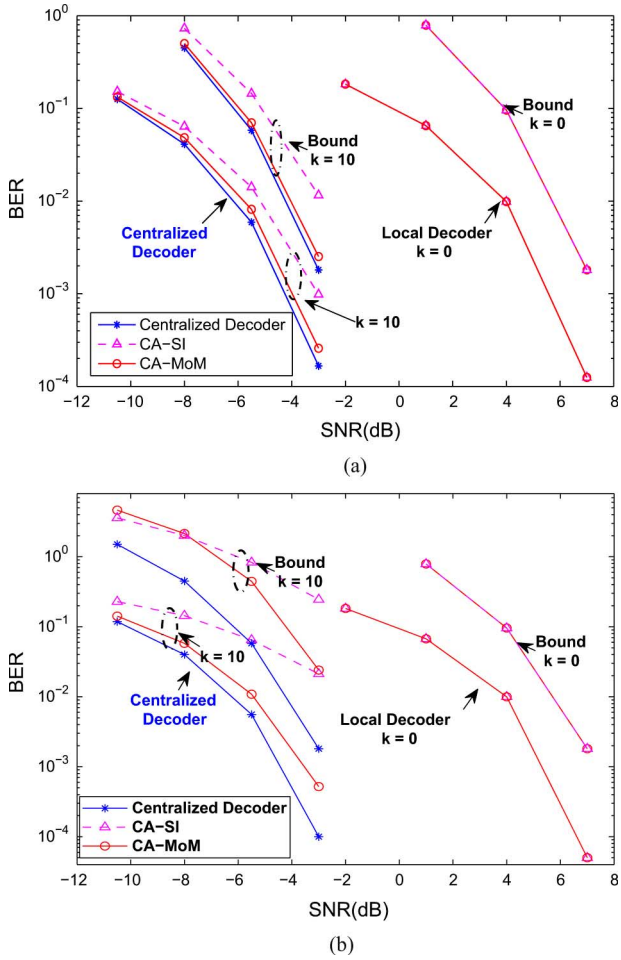


Fig. 5. Distributed ML decoder performance using $(7,4)$ Hamming codes. BER versus SNR (in dB) comparison of error bounds with: (a) Ideal inter-sensor links; (b) Noisy inter-sensor links.

of the CA-MoM algorithm and the CA-SI increases markedly, since the CA-SI has to resort to a vanishing stepsize to ensure convergence even for a relatively low inter-sensor noise level of 20 dB. Interestingly, even under these adverse conditions, the CA-MoM algorithm after $k = 10$ iterations exhibits approximately the same performance as the CA-SI one for ideal AP-sensor links, and is still close to the centralized one that is by definition not affected by inter-sensor imperfections with all observations available centrally. Fig. 4(b) depicts the performance of various distributed decoders given different inter-sensor link conditions. The AP-sensor SNR level is set to be -1 dB. The inter-sensor noise level affects the convergence floor of the distributed decoders; while the link failure probability determines the convergence rate. For example, with $p_{ji} = 0.9$, both distributed CA-MoM decoders reach steady state after 30 iterations approximately. Intuitively, a lower inter-sensor noise level $(\mathrm{SNR} = 20$ dB$)$ will lead to a much better BER performance. In addition, given the same inter-sensor link SNR level of 20 dB, the CA-MoM decoder with $p_{ji} = 0.9$ converges much faster. Intuitively, the larger p_{ji} is, the less probable it is for each link to fail per transmission, and in this sense the CA-MoM decoder operates more efficiently.

Test Case 4: (*BER bounds*) Here, the error performance bounds developed in Section V are tested for block ML decoding of the Hamming (7,4) code. Both ideal and noisy inter-sensor links are assumed. The theoretical upper bounds on the BER per iteration are computed based on (36). Fig. 5(a) and (b) plot the resulting curves for different values of k with ideal and noisy inter-sensor links, respectively. The simulated BER obtained through Monte Carlo runs is also included for comparison. The theoretical upper bounds exhibit the same diversity order as the simulated BER, verifying the validity of $\Delta_j^{(\lambda)}(k)$ in (33) as a performance metric of the CA-SI based block ML decoders. In practical distributed decoding problems, these upper bounds offer a useful means of determining the number of iterations needed to guarantee a given level of error performance at any given sensor.

VII. Conclusion

The problem of distributed block- and bitwise- decoding of a coded message transmitted from an AP to a WSN was considered. Sufficient statistics for decoding were expressed in terms of the average of local LLRs. Based on the latter, two algorithms were presented to obtain such sufficient statistics in a distributed fashion: i) the CA-SI scheme, which relies on weighted iterations of local averages; and ii) the CA-MoM scheme, which solves a distributed optimization problem with the alternating direction MoM. Unlike distributed detection alternatives that incur exponential complexity, both have complexity that scales linearly with the problem dimension. With suitable modifications, both algorithms were made robust to random link failures and additive noise present in the inter-sensor links. Error performance bounds were also derived to determine the number of iterations needed to guarantee a given level of average decoding errors at any given sensor. Simulations demonstrated that after a few iterations, the BER of both CA-SI and CA-MoM algorithms come very close to that of their centralized benchmark.

Appendix A
Sufficient Statistics for Arbitrary Alphabets

If x_n takes values from the alphabet $\{0,\ldots,Q-1\}$, then $\log p[y_{jn}|x_n]$ is a piece-wise linear function of x_n [c.f. (4)]. Specifically, if $x_n \in \{q-1, q\}$, then $\log p[y_{jn}|x_n]$ is given by

$$\log p[y_{jn}|x_n] = -(x_n - q + 1)\log\left(\frac{p[y_{jn}|x_n = q-1]}{p[y_{jn}|x_n = q]}\right) + \log p[y_{jn}|x_n = q-1]. \quad (37)$$

Because the sum of piece-wise linear functions is a piece-wise linear function, the ML decoding problem in (1) is equivalent to

$$\hat{\mathbf{x}}_{ML} = \arg\min_{\mathbf{x}\in\mathcal{C}} \sum_{n=1}^{N} \gamma_n^{PL}(x_n) \quad (38)$$

where

$$\gamma_n^{PL}(x_n) := (x_n - q + 1)\left(\frac{1}{J}\sum_{j=1}^{J}\gamma_{jn}^{(q)}\right) + \sum_{q'=1}^{q-1}\left(\frac{1}{J}\sum_{j=1}^{J}\gamma_{jn}^{(q')}\right) \quad (39)$$

for $x_n \in \{q-1, q\}$, and $q = 1, \ldots, Q-1$. With the local LLRs defined as

$$\gamma_{jn}^{(q)} = \left(\frac{p[y_{jn}|x_n = q-1]}{p[y_{jn}|x_n = q]} \right), \quad q = 1, \ldots, Q-1 \tag{40}$$

it follows that $(Q-1)$ average LLRs are sufficient statistics for this distributed decoding problem.

APPENDIX B
DERIVATION OF (18a)–(18b)

Because the variables $\bar{\gamma}_{jn}$ in (16) are not coupled, (17a) is equivalent to the following J separable sub-problems, one per sensor j

$$\bar{\gamma}_{jn}(k+1) = \arg\min_{\bar{\gamma}_{jn}} \Bigg\{ \frac{1}{2}(\gamma_{jn} - \bar{\gamma}_{jn})^2 + \sum_{i \in \mathcal{N}_j} (v_{jin}(k) - v'_{ijn}(k))\, \bar{\gamma}_{jn} + \sum_{i \in \mathcal{N}_j} \frac{c}{2} \left[(\bar{\gamma}_{jn} - z_{jin}(k))^2 + (-\bar{\gamma}_{jn} - z'_{ijn}(k))^2 \right] \Bigg\}. \tag{41}$$

Being linear-quadratic in $\bar{\gamma}_{jn}$, each of these sub-problems can be solved in closed form to obtain

$$\bar{\gamma}_{jn}(k+1) = \frac{1}{1 + 2c|\mathcal{N}_j|} \Bigg\{ \gamma_{jn} - \sum_{i \in \mathcal{N}_j} [v_{jin}(k) - v'_{ijn}(k) - c(z_{jin}(k) - z'_{ijn}(k))] \Bigg\}. \tag{42}$$

Similarly, $z_n(k+1)$ in (17b) is obtained by solving the following $2|\mathcal{E}|$ sub-problems indexed by $(j,i) \in \mathcal{E}$

$$\{z_{jin}(k+1), z'_{jin}(k+1)\} = \arg \min_{z_{jin} + z'_{jin} = 0} \Bigg\{ - v_{jin}(k) z_{jin} - v'_{jin}(k) z'_{jin} + \frac{c}{2}(\bar{\gamma}_{jn}(k+1) - z_{jin})^2 + \frac{c}{2}(-\bar{\gamma}_{jn}(k+1) - z'_{jin})^2 \Bigg\}. \tag{43}$$

If $v_{jin}(k)$ and $v'_{jin}(k)$ are initialized as $v_{jin}(0) = v'_{jin}(0)\ \forall (j,i) \in \mathcal{E}$, the solution to (43) for $k = 0$ becomes

$$z_{jin}(1) = -z'_{jin}(1) = \frac{1}{2}(\bar{\gamma}_{jn}(1) + \bar{\gamma}_{in}(1)). \tag{44}$$

Substituting (44) into (17c) and (17d) yields

$$v_{jin}(1) = v'_{jin}(1) = v_{jin}(0) + \frac{c}{2}(\bar{\gamma}_{jn}(1) - \bar{\gamma}_{in}(1)). \tag{45}$$

Proceeding by induction, if $v_{jin}(k) = v'_{jin}(k)$, the solution to (43) is

$$z_{jin}(k+1) = -z'_{jin}(k+1) = \frac{1}{2}(\bar{\gamma}_{jn}(k+1) + \bar{\gamma}_{in}(k+1)). \tag{46}$$

Substituting (46) into (17c) and (17d) proves that

$$v_{jin}(k+1) = v'_{jin}(k+1) = v_{jin}(k) + \frac{c}{2}(\bar{\gamma}_{jn}(k+1) - \bar{\gamma}_{in}(k+1)) \tag{47}$$

which establishes along with (45) that $v_{jin}(k) = v'_{jin}(k)\ \forall k > 0$.

Equality (47) shows that it is sufficient to update only one set of multipliers $\{v_{jin}(k)\}_{j \in \mathcal{J}}^{i \in \mathcal{N}_j}$ per iteration as in (18a). Finally, substituting (46) into (42) proves the validity of (18b).

APPENDIX C
PROOF OF PROPOSITION 3

To prove convergence in the mean, recall that the noise terms $\eta_{jin}(k)$ and $\bar{\eta}_{jin}(k)$ in (20a) and (20b) are zero-mean and uncorrelated across time and links. Given the linearity of the expectation operator, taking averages on both sides of (20a) and (20b) results in iterations similar to (18a) and (18b), which involve deterministic iterates whose convergence is ensured by Proposition 2.

To establish that the iterates in (20a)–(20b) have bounded variance, define the $2|\mathcal{E}| \times 1$ vectors $\boldsymbol{\eta}_n(k) := [\{\eta_{1in}(k)\}_{i \in \mathcal{N}_1}, \ldots, \{\eta_{Jin}(k)\}_{i \in \mathcal{N}_j}]^T$ and $\bar{\boldsymbol{\eta}}_n(k) := [\{\bar{\eta}_{1in}(k)\}_{i \in \mathcal{N}_1}, \ldots, \{\bar{\eta}_{Jin}(k)\}_{i \in \mathcal{N}_j}]^T$; and the $J \times 2|\mathcal{E}|$ incidence matrix $\mathbf{M}$, see, e.g., [20, p. 56], formed by columns $\{\mathbf{m}_p\}_{p=1}^{2|\mathcal{E}|}$ of size $J \times 1$. With the subscript p corresponding to the $j \rightarrow i$ directed edge of the graph, the $\mathbf{m}_{p(j \rightarrow i)}$ column has all entries equal to 0 except its jth and ith, for which $[\mathbf{m}_{p(j \rightarrow i)}]_j = 1$ and $[\mathbf{m}_{p(j \rightarrow i)}]_i = -1$. Using the so defined matrix $\mathbf{M}$, (20a) can be expressed in vector form as

$$\mathbf{v}_n(k) = \mathbf{v}_n(k-1) + \frac{c}{2}[\mathbf{M}^T \bar{\boldsymbol{\gamma}}_n(k) - \bar{\boldsymbol{\eta}}_n(k)]. \tag{48a}$$

Let now $\mathbf{M}'$ denote the matrix formed by setting to zero all non-positive entries of $\mathbf{M}$; and $\mathbf{N}_c := \mathbf{I}_J + 2c\mathbf{D}$, where $\mathbf{D}$ is the degree matrix defined in the opening paragraph of Section II. Using these definitions, (20b) can be likewise expressed in vector form as

$$\bar{\boldsymbol{\gamma}}_n(k+1) = \mathbf{N}_c^{-1}[\bar{\boldsymbol{\gamma}}_n - \mathbf{M}\mathbf{v}_n(k) + c(\mathbf{A} + \mathbf{D})\bar{\boldsymbol{\gamma}}_n(k) + \mathbf{M}'\boldsymbol{\eta}_n(k) + c\mathbf{M}'\bar{\boldsymbol{\eta}}_n(k)]. \tag{48b}$$

Upon defining the $(J + 2|\mathcal{E}|) \times 1$ augmented state vector $\mathbf{s}_n^a(k) := [\bar{\boldsymbol{\gamma}}_n^T(k+1), \mathbf{v}_n^T(k)]^T$, the iterations (48a) and (48b) can be collectively written as

$$\mathbf{s}_n^a(k) = \mathbf{W}^a \mathbf{s}_n^a(k-1) + \mathbf{b}_n^a + \boldsymbol{\eta}_n^a(k) \tag{49}$$

where the $(J + 2|\mathcal{E}|) \times (J + 2|\mathcal{E}|)$ augmented state transition matrix $\mathbf{W}^a$, and the $(J + 2|\mathcal{E}|) \times 1$ vector $\mathbf{b}_n^a$ are given, respectively, by

$$\mathbf{W}^a := \begin{bmatrix} 2c\mathbf{N}_c^{-1}\mathbf{A} & -\mathbf{N}_c^{-1}\mathbf{M} \\ \frac{c}{2}\mathbf{M}^T & \mathbf{I}_{2|\mathcal{E}|} \end{bmatrix}, \text{ and } \mathbf{b}_n^a := \begin{bmatrix} \mathbf{N}_c^{-1}\boldsymbol{\gamma}_n \\ \mathbf{0}_{2|\mathcal{E}|} \end{bmatrix} \tag{50}$$

whereas the augmented noise term $\boldsymbol{\eta}_n^a(k)$ can be expressed as $\boldsymbol{\eta}_n^a(k) := \mathbf{E}\boldsymbol{\eta}_n(k) + \bar{\mathbf{E}}\bar{\boldsymbol{\eta}}_n(k)$ with

$$\mathbf{E} := \begin{bmatrix} \mathbf{N}_c^{-1}\mathbf{M}' \\ \mathbf{0}_{2|\mathcal{E}|} \end{bmatrix}, \text{ and } \bar{\mathbf{E}} := \begin{bmatrix} \frac{c}{2}\mathbf{N}_c^{-1}(\mathbf{M}+2\mathbf{M}') \\ \frac{-c}{2}\mathbf{I}_{2|\mathcal{E}|} \end{bmatrix}. \quad (51)$$

The next lemma establishes useful properties for the eigenvalues and eigenvectors of $\mathbf{W}^a$.

Lemma 2: *If $c > 0$, and the eigenvalues $\{\lambda_i\}_{i=1}^{J+2|\mathcal{E}|}$ of $\mathbf{W}^a$ are ordered so that $|\lambda_1| \leq \cdots \leq |\lambda_{J+2|\mathcal{E}|}|$, then:*

(a) *$|\lambda_i| < 1$ for $i = 1, 2, \ldots, 2J-1$, and $\lambda_i = 1$ for $i = 2J, \ldots, J+2|\mathcal{E}|$;*

(b) *with $\mathbf{u}_{1,i}$ ($\mathbf{w}_{1,i}$) representing the first J entries of the right (left) eigenvector of $\mathbf{W}^a$, and $\mathbf{u}_{2,i}$ ($\mathbf{w}_{2,i}$) the last $2|\mathcal{E}|$ entries, it holds that:*

$$\mathbf{u}_{1,i} = \mathbf{w}_{1,i} = \mathbf{M}\mathbf{u}_{2,i} = \mathbf{M}\mathbf{w}_{2,i} = \mathbf{0}_J, \quad i = 2J, \ldots, J+2|\mathcal{E}|. \quad (52)$$

Proof: Since $\mathbf{W}^a\mathbf{u}_i = \lambda_i\mathbf{u}_i$, it follows that $\mathbf{W}_1^a\mathbf{u}_i = \lambda_i\mathbf{u}_{1,i}$ and $\mathbf{W}_2^a\mathbf{u}_i = \lambda_i\mathbf{u}_{2,i}$, where $\mathbf{W}_1^a$ ($\mathbf{W}_2^a$) is formed by the first (last) J ($2|\mathcal{E}|$) rows of $\mathbf{W}^a$. Using (50) to eliminate $\mathbf{u}_{2,i}$ from these eigenvector-eigenvalue equations, and left multiplying the result with $\mathbf{u}_{1,i}^{\mathcal{H}}$, yields the second-order equation

$$\left(\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{N}_c\mathbf{u}_{1,i}\right)\lambda_i^2 - \left[\mathbf{u}_{1,i}^{\mathcal{H}}(\mathbf{I}+2c\mathbf{A}+2c\mathbf{D})\mathbf{u}_{1,i}\right]\lambda_i + c\mathbf{u}_{1,i}^{\mathcal{H}}(\mathbf{A}+\mathbf{D})\mathbf{u}_{1,i} = 0, \quad (53)$$

whose roots can be expressed as $\lambda_i = \alpha_1 \pm \sqrt{\alpha_1^2 - \alpha_2}$, where

$$\alpha_1 := \frac{\mathbf{u}_{1,i}^{\mathcal{H}}(\mathbf{I}+2c\mathbf{A}+2c\mathbf{D})\mathbf{u}_{1,i}}{2\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{N}_c\mathbf{u}_{1,i}}, \quad \text{and}$$
$$\alpha_2 := \frac{c\mathbf{u}_{1,i}^{\mathcal{H}}(\mathbf{A}+\mathbf{D})\mathbf{u}_{1,i}}{\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{N}_c\mathbf{u}_{1,i}}. \quad (54)$$

Because $c > 0$, for any non-zero $\mathbf{u}_{1,i}$, it holds that

$$2c\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{A}\mathbf{u}_{1,i} \leq 2c\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{D}\mathbf{u}_{1,i} < \mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{N}_c\mathbf{u}_{1,i} \quad (55)$$

where the first inequality comes from the positive semi-definiteness of the graph Laplacian $\mathbf{L} := \mathbf{D} - \mathbf{A}$ [20, p. 469], and the second is due to the definition $\mathbf{N}_c := \mathbf{I}_J + 2c\mathbf{D}$.

If $\alpha_1^2 < \alpha_2$, then the complex root λ_i has nonzero imaginary part. Using (54) and (55), one can deduce that in this case $|\lambda_i| = \alpha_2 < 1$. If $\alpha_1^2 \geq \alpha_2$, then λ_i is real. In this case, (55) implies that $\alpha_1 < 1$. Upon defining $\alpha_3 := 1 - \alpha_1 = \mathbf{u}_{1,i}^{\mathcal{H}}(\mathbf{I}+2c\mathbf{L})\mathbf{u}_{1,i}/(2\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{N}_c\mathbf{u}_{1,i})$, the root can be rewritten as

$$\lambda_i = 1 - \alpha_3 \pm \sqrt{\alpha_3^2 - \frac{c\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{L}\mathbf{u}_{1,i}}{\mathbf{u}_{1,i}^{\mathcal{H}}\mathbf{N}_c\mathbf{u}_{1,i}}}. \quad (56)$$

Using again (55), it follows that $0 \leq \alpha_3 < 1$, so the maximum value inside the square root is α_3^2; thus, $|\lambda_i| \leq 1$. Strict equality holds when $\mathbf{L}\mathbf{u}_{1,i} = \mathbf{0}_J$. In this case, the graph connectivity of (a1) implies that $\mathbf{u}_{1,i} = u_{1,i}\mathbf{1}_J$. Substituting into $\mathbf{W}^a\mathbf{u}_i = \lambda_i\mathbf{u}_i$ gives $u_{1,i}\mathbf{1}_J + \mathbf{M}\mathbf{u}_{2,i} = \mathbf{0}_J$; and left multiplying the latter with $\mathbf{1}_J^T$ yields $u_{1,i} = 0$, and thus $\mathbf{M}\mathbf{u}_{2,i} = \mathbf{0}_J$. The connectivity of the graph further implies that $\mathrm{rank}(\mathbf{M}) = J-1$; hence, the geometric multiplicity of the eigenvalue $\lambda_i = 1$, equal to the nullity of $\mathbf{M}$, is $2|\mathcal{E}| - J + 1$. Similar reasoning applied to the left eigenvectors implies that $\mathbf{w}_{1,i} = \mathbf{M}\mathbf{w}_{2,i} = \mathbf{0}_J$ corresponding to $\lambda_i = 1$. ■

To proceed with the proof of Proposition 3, note that (49) can be rewritten as

$$\mathbf{s}_n^a(k) = (\mathbf{W}^a)^k\mathbf{s}_n^a(0) + \sum_{l=0}^{k-1}(\mathbf{W}^a)^l[\mathbf{b}_n^a + \boldsymbol{\eta}_n^a(k-l)] \quad (57)$$

from where one can readily obtain the covariance matrix of $\mathbf{s}_n^a(k)$ as

$$\boldsymbol{\Sigma}_{s_n}(k) := E\left\{\left[\mathbf{s}_n^a(k) - E\mathbf{s}_n^a(k)\right]\left[\mathbf{s}_n^a(k) - E\mathbf{s}_n^a(k)\right]^T\right\} = E\left\{\left[\sum_{l=0}^{k-1}(\mathbf{W}^a)^l\boldsymbol{\eta}_n^a(k-l)\right]\left[\sum_{l=0}^{k-1}(\mathbf{W}^a)^l\boldsymbol{\eta}_n^a(k-l)\right]^T\right\}. \quad (58)$$

Since $(\mathbf{W}^a)^l = \sum_{i=1}^{J+2|\mathcal{E}|}\lambda_i^l\mathbf{u}_i\mathbf{w}_i^T$, and $\boldsymbol{\eta}_n^a(k)$ is uncorrelated across time, (58) reduces to

$$\boldsymbol{\Sigma}_{s_n}(k) = \sum_{l=0}^{k-1}\left(\sum_{i=1}^{J+2|\mathcal{E}|}\lambda_i^l\mathbf{u}_i\mathbf{w}_i^T\right)\boldsymbol{\Sigma}_{\eta_n^a}\left(\sum_{i=1}^{J+2|\mathcal{E}|}\lambda_i^l\mathbf{w}_i\mathbf{u}_i^T\right) = \sum_{i=1}^{J+2|\mathcal{E}|}\sum_{i'=1}^{J+2|\mathcal{E}|}\left(\sum_{l=0}^{k-1}(\lambda_i\lambda_{i'})^l\right)(\mathbf{w}_i^T\boldsymbol{\Sigma}_{\eta_n^a}\mathbf{w}_{i'})\mathbf{u}_i\mathbf{u}_{i'}^T \quad (59)$$

where $\boldsymbol{\Sigma}_{\eta_n^a}$ is given by

$$\boldsymbol{\Sigma}_{\eta_n^a} := E\left\{\boldsymbol{\eta}_n^a(k)\left(\boldsymbol{\eta}_n^a(k)\right)^T\right\} = \mathbf{E}\boldsymbol{\Sigma}_{\eta_n}\mathbf{E}^T + \bar{\mathbf{E}}\boldsymbol{\Sigma}_{\bar{\eta}_n}\bar{\mathbf{E}}^T \quad (60)$$

with $\boldsymbol{\Sigma}_{\eta_n}$ and $\boldsymbol{\Sigma}_{\bar{\eta}_n}$ denoting the covariance matrices of $\boldsymbol{\eta}_n(k)$ and $\bar{\boldsymbol{\eta}}_n(k)$, respectively. Clearly, the diagonal entries of both matrices are σ_j^2 with the subscript j depending on the receive sensor index for the corresponding noise $\eta_{jin}(k)$ in $\boldsymbol{\eta}_n(k)$, or, $\bar{\eta}_{jin}(k)$ in $\bar{\boldsymbol{\eta}}_n(k)$.

Recalling the definition of $\mathbf{s}_n^a(k)$, the covariance $\boldsymbol{\Sigma}_{\bar{\gamma}_n}(k) := E[\bar{\boldsymbol{\gamma}}_n(k)\bar{\boldsymbol{\gamma}}_n^T(k)]$ is the $J \times J$ upper-left sub-block of $\boldsymbol{\Sigma}_{s_n}(k-1)$; hence after using (52) and (59) implies

$$\boldsymbol{\Sigma}_{\bar{\gamma}_n}(k+1) = \sum_{i=1}^{2J-1}\sum_{i'=1}^{2J-1}\left(\sum_{l=0}^{k-1}(\lambda_i\lambda_{i'})^l\right)(\mathbf{w}_i^T\boldsymbol{\Sigma}_{\eta_n^a}\mathbf{w}_{i'})\mathbf{u}_{i,1}\mathbf{u}_{i',1}^T. \quad (61)$$

Lemma 2(a) ensures that $|\lambda_i\lambda_{i'}| < 1$ for $i, i' = 1, 2, \ldots, 2J-1$, leading to

$$\lim_{k\to\infty}\boldsymbol{\Sigma}_{\bar{\gamma}_n}(k) = \sum_{i=1}^{2J-1}\sum_{i'=1}^{2J-1}\frac{\mathbf{w}_i^T\boldsymbol{\Sigma}_{\eta_n^a}\mathbf{w}_{i'}}{1-\lambda_i\lambda_{i'}}\mathbf{u}_{i,1}\mathbf{u}_{i',1}^T \quad (62)$$

which shows that the covariance of CA-MoM iterates is bounded and completes the proof of Proposition 3.

APPENDIX D
PROOF OF PROPOSITION 4

Convergence in this case will be established through a non-increasing Lyapunov function [1, Sec. 6.10], of the iterates (22a)–(22c). With this objective in mind, and similar

to Appendix C, we first express the iterations (22a)–(22c) in vector form

$$\mathbf{v}_n(k) = \mathbf{v}_n(k-1) + \frac{c}{2}\mathbf{M}^T(k)\bar{\boldsymbol{\gamma}}_n(k) \tag{63a}$$

$$\bar{\boldsymbol{\gamma}}_n(k+1) = \mathbf{N}_c^{-1}\Big\{\boldsymbol{\gamma}_n - \bar{\mathbf{M}}\mathbf{v}_n(k) + c[2\mathbf{D} - \mathbf{L}(k)]\bar{\boldsymbol{\gamma}}_n(k)\Big\} \tag{63b}$$

where $\mathbf{M}(k)$ is the $J \times 2|\mathcal{E}|$ time-varying incidence matrix and $\bar{\mathbf{M}} := E\{\mathbf{M}(k)\}$ is its expected value. Notice that $\bar{\mathbf{M}}$ has the same non-zero entries as $\mathbf{M}$, with each entry replaced by the corresponding link failure probability p_{ji} as prescribed in (a2.3). Next, define the limits $\bar{\boldsymbol{\gamma}}_n^* := \bar{\gamma}_n\mathbf{1}$ and $\mathbf{v}_n^*$ such that $\bar{\mathbf{M}}\mathbf{v}_n^* := \bar{\boldsymbol{\gamma}}_n - \bar{\boldsymbol{\gamma}}_n^*$. If the network is connected, matrix $\bar{\mathbf{M}}^T$ has rank $J-1$ and its null space is $\{\mu\mathbf{1}_J\}$, $\forall\mu \in \mathbb{R}$ since $\bar{\mathbf{M}}^T\mathbf{1}_J = \mathbf{0}_{2|\mathcal{E}|}$ (note that each column $\bar{\mathbf{m}}_{p(j\longrightarrow i)}$ of $\bar{\mathbf{M}}$ has only two non-zero entries $\pm p_{ji}$). This ensures the existence of vector $\mathbf{v}_n^*$ since $\bar{\boldsymbol{\gamma}}_n - \bar{\boldsymbol{\gamma}}_n^*$ is guaranteed to be in the range space of $\bar{\mathbf{M}}$. Therefore, the equivalent recursions for the difference vectors $\tilde{\boldsymbol{\gamma}}_n(k) := \bar{\boldsymbol{\gamma}}_n(k) - \bar{\boldsymbol{\gamma}}_n^*$ and $\tilde{\mathbf{v}}_n(k) := \mathbf{v}_n(k) - \mathbf{v}_n^*$ become [c.f. (63a)–(63b)]

$$\tilde{\mathbf{v}}_n(k) = \tilde{\mathbf{v}}_n(k-1) + \frac{c}{2}\mathbf{M}^T(k)\tilde{\boldsymbol{\gamma}}_n(k) \tag{64a}$$

$$\tilde{\boldsymbol{\gamma}}_n(k+1) = \mathbf{N}_c^{-1}\Big\{-\bar{\mathbf{M}}\tilde{\mathbf{v}}_n(k) + c[2\mathbf{D} - \mathbf{L}(k)]\tilde{\boldsymbol{\gamma}}_n(k)\Big\}. \tag{64b}$$

We wish to prove that the iterations (64a) and (64b) reach an equilibrium as $k \longrightarrow \infty$. To this end, consider the Lyapunov function

$$\begin{aligned}\mathcal{V}(k) :=& 2\|\tilde{\mathbf{v}}_n(k)\|^2 + 2c\tilde{\mathbf{v}}_n^T(k)\bar{\mathbf{M}}\tilde{\boldsymbol{\gamma}}_n(k+1) \\ &+ 2c^2\tilde{\boldsymbol{\gamma}}_n^T(k+1)\mathbf{D}\tilde{\boldsymbol{\gamma}}_n(k+1) \\ =& 2\|\tilde{\mathbf{v}}_n(k) + \frac{c}{2}\bar{\mathbf{M}}\tilde{\boldsymbol{\gamma}}_n(k+1)\|^2 \\ &+ \frac{c^2}{2}\tilde{\boldsymbol{\gamma}}_n^T(k+1)(4\mathbf{D} - \bar{\mathbf{M}}^T\bar{\mathbf{M}})\tilde{\boldsymbol{\gamma}}_n(k+1). \end{aligned} \tag{65}$$

The positive semi-definiteness of matrix $(4\mathbf{D} - \bar{\mathbf{M}}^T\bar{\mathbf{M}})$ ensures that $\mathcal{V}(k)$ is a valid (positive) candidate Lyapunov function of $(\tilde{\boldsymbol{\gamma}}_n(k+1), \tilde{\mathbf{v}}_n(k))$. Substituting (64a) and (64b) into (65) the following difference of successive Lyapunov functions becomes:

$$\begin{aligned}&\mathcal{V}(k-1) - \mathcal{V}(k) \\ &\quad= 2c\|\tilde{\boldsymbol{\gamma}}_n(k+1)\|^2 + 2c\tilde{\mathbf{v}}_n^T(k-1)[\bar{\mathbf{M}} - \mathbf{M}(k)]\tilde{\boldsymbol{\gamma}}_n(k) \\ &\quad\quad+ c^2\tilde{\boldsymbol{\gamma}}_n^T(k+1)\mathbf{L}(k)\tilde{\boldsymbol{\gamma}}_n(k+1) + c^2(\tilde{\boldsymbol{\gamma}}_n(k+1) - \tilde{\boldsymbol{\gamma}}_n(k))^T \\ &\quad\quad(2\mathbf{D} - \mathbf{L}(k))(\tilde{\boldsymbol{\gamma}}_n(k+1) - \tilde{\boldsymbol{\gamma}}_n(k)).\end{aligned} \tag{66}$$

Because the zero-mean matrix difference $\bar{\mathbf{M}} - \mathbf{M}(k)$ is uncorrelated with the state $(\tilde{\boldsymbol{\gamma}}_n(k), \tilde{\mathbf{v}}_n(k-1))$ at the previous iteration [c.f. (a2.3)], the expectation of (65) satisfies (since the last two quadratic terms in (65) are positive semi-definite)

$$E\mathcal{V}(k-1) - E\mathcal{V}(k) \geq 2cE\{\|\tilde{\boldsymbol{\gamma}}_n(k+1)\|^2\}. \tag{67}$$

Summing up the first K of these differences, yields

$$\sum_{k=1}^{K} E\{\|\tilde{\boldsymbol{\gamma}}_n(k)\|^2\} \leq \frac{[E\mathcal{V}(0) - E\mathcal{V}(K-1)]}{2c} \leq \frac{E\mathcal{V}(0)}{2c} \tag{68}$$

where the last inequality comes from the non-negativity of $\mathcal{V}(K-1)$. The partial sum of the non-negative series $\{E\{\|\tilde{\boldsymbol{\gamma}}_n(k)\|^2\}\}_{k=1}^{\infty}$ in (68) is bounded; thus, $E\{\|\tilde{\boldsymbol{\gamma}}_n(k)\|^2\}$ converges to zero as $k \longrightarrow \infty$, and so does $E\{|\tilde{\boldsymbol{\gamma}}_{jn}(k)|^2\}$, which completes the proof of (23).

Appendix E
Proof of Lemma 1

Consider first the CA-SI in (11), which in vector form can be expressed after backward substitution as $\bar{\boldsymbol{\gamma}}_n(k) = \mathbf{W}^k\bar{\boldsymbol{\gamma}}_n(0)$. Writing the latter entry-wise proves (32) after setting $c_{ji}(k) = [\mathbf{W}^k]_{ji}$.

To prove (32) for the CA-MoM iterations in (18a)–(18b), consider as in Appendix C the augmented vector recursion

$$\mathbf{s}_n^a(k) = \mathbf{W}^a\mathbf{s}_n^a(k-1) + \mathbf{b}_n^a \tag{69}$$

where $\mathbf{W}^a$ and $\mathbf{b}_n^a$ are as in (50). Initializing with $\mathbf{s}_n^a(0) := [\boldsymbol{\gamma}_n^T\ \mathbf{0}_{2|\mathcal{E}|}^T]^T$, backward substitution yields

$$\begin{aligned}\mathbf{s}_n^a(k) &= (\mathbf{W}^a)^k\,\mathbf{s}_n^a(0) + \sum_{l=0}^{k-1}(\mathbf{W}^a)^l\mathbf{b}_n^a \\ &= \sum_{i=1}^{J+2|\mathcal{E}|}\lambda_i^k s_i\mathbf{u}_i + \sum_{l=0}^{k-1}\sum_{i=1}^{J+2|\mathcal{E}|}\lambda_i^l b_i\mathbf{u}_i\end{aligned} \tag{70}$$

where

$$\begin{aligned}&s_i := \mathbf{w}_{i,1}^T\boldsymbol{\gamma}_n,\ b_i = \mathbf{w}_{i,1}^T\mathbf{N}_c^{-1}\boldsymbol{\gamma}_n,\ i = 1,2,\ldots,2J-1 \\ &s_i = b_i = 0, \quad i = 2J,\ldots,2J+|\mathcal{E}|.\end{aligned} \tag{71}$$

With these definitions, the upper part of (70) is given by

$$\begin{aligned}\bar{\boldsymbol{\gamma}}_n(k) &= \sum_{i=1}^{2J-1}\lambda_i^k\mathbf{u}_{i,1}\mathbf{w}_{i,1}^T\boldsymbol{\gamma}_n + \sum_{l=0}^{k-1}\sum_{i=1}^{2J-1}\lambda_i^l\mathbf{u}_{i,1}\mathbf{w}_{i,1}^T\mathbf{N}_c^{-1}\boldsymbol{\gamma}_n \\ &= \sum_{i=1}^{2J-1}\left(\lambda_i^k\mathbf{u}_{i,1}\mathbf{w}_{i,1}^T + \frac{1-\lambda_i^k}{1-\lambda_i}\mathbf{u}_{i,1}\mathbf{w}_{i,1}^T\mathbf{N}_c^{-1}\right)\boldsymbol{\gamma}_n.\end{aligned} \tag{72}$$

Written entry-wise, (72) can be expressed as in (32), which proves the lemma for the CA-MoM iterates, provided that one initializes (18a) and (18b) with $[\boldsymbol{\gamma}_n^T, \mathbf{0}^T]^T$.

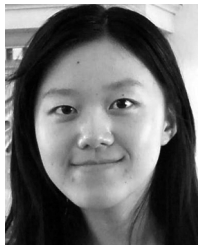
References

[1] P. J. Antsaklis and A. N. Michel, *Linear Systems*. New York: McGraw-Hill, 1997.

[2] T. C. Aysal and K. E. Barner, "On the convergence of perturbed and non-stationary consensus algorithms," in *Proc. IEEE INFOCOM*, Rio de Janeiro, Brazil, Apr. 19–25, 2009.

[3] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Belmont, MA: Athena Scientific, 1997.

[4] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Belmont, MA: Athena Scientfic, 1999.
[5] J. Feldman, M. J. Wainwright, and D. R. Karger, "Using linear programming to decode binary linear codes," *IEEE Trans. Inf. Theory*, vol. 51, pp. 954–972, Mar. 2005.
[6] Y. Hatano, A. K. Das, and M. Mesbahi, "Agreement in presence of noise: Pseudogradients on random geometric networks," in *Proc. 44th Conf. Decision Control Eur. Control Conf.*, Seville, Spain, Dec. 2005, pp. 6382–6387.
[7] A. Jadbabaie, J. Lin, and S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Autom. Control*, vol. 48, pp. 988–1001, Jun. 2003.
[8] S. Kar and J. M. F. Moura, "Distributed consensus algorithms in sensor networks with communication channel noise and random link failures," in *Proc. 41st Asilomar Conf. Signals, Systems, Computers*, Pacific Grove, CA, Oct. 2007.
[9] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Control*, vol. 49, pp. 1520–1533, Sep. 2004.
[10] R. Olfati-Saber, E. Frazzoli, E. Franco, and J. S. Shamma, "Belief consensus and distributed hypothesis testing in sensor networks," in *Network Embedded Sensing and Control*, P. J. Antsaklis, P. Tabuada, and Editors, Eds. New York: Springer-Verlag, 2006, pp. 169–182.
[11] B. T. Polyak and Y. Z. Tsypkin, "Pseudogradient adaptation and training algorithms," *Autom. Remote Control*, no. 3, pp. 45–68, 1973.
[12] J. G. Proakis and M. Salehi, *Digital Communications*, fifth ed. New York: McGraw-Hill, 2008.
[13] M. G. Rabbat, R. D. Nowak, and J. A. Bucklew, "Generalized consensus algorithms in networked systems with erasure links," in *Proc. Signal Process. Advances Wireless Commun.*, New York, Jun. 2005.
[14] R. Rajagopal and M. J. Wainwright, "Network-based consensus averaging with general noisy channels," in *Proc. 45th Allerton Conf. Commun., Control, and Comput.*, Monticello, IL, Sep. 26–28, 2007.
[15] A. Ribeiro, I. D. Schizas, J.-J. Xiao, G. B. Giannakis, and Z.-Q. Luo, "Distributed estimation under bandwidth and energy constraints," in *Wireless Sensor Networks: Signal Processing and Communicaitons Perspectives*, A. Swami, Q. Zhao, Y. Hong, and L. Tong, Eds. New York: Wiley, 2007, ch. 7, pp. 149–184.
[16] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge, U.K.: Cambridge Univ. Press, 2007.
[17] V. Saligrama, M. Alanyali, and O. Savas, "Distributed detection in sensor networks with packet losses and finite capacity links," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4118–4132, Nov. 2006.
[18] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, "Consensus in ad hoc WSNs with noisy links—Part I: Distributed estimation of deterministic signals," *IEEE Trans. Signal Process.*, vol. 56, no. 1, pp. 342–356, Jan. 2008.
[19] V. Solo and X. Kong, *Adaptive Signal Processing Algorithms: Stability and Performance*. Englewood Cliffs, NJ: Prentice-Hall, 1995.
[20] D. B. West, *Introduction to Graph Theory*, Second ed. Englewood Cliffs, NJ: Prentice-Hall, 2000.
[21] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Syst. Control Lett.*, vol. 53, pp. 65–78, Sep. 2004.
[22] S. Zenios and Y. Censor, *Parallel Optimization: Theory, Algorithms and Applications*. New York: Oxford Univ. Press, 1997.



Hao Zhu (S'07) received the Bachelor's degree in information electronics and engineering from the Department of Electronic Engineering, Tsinghua University, Beijing, China, in 2006. Since September 2006, she has been working towards the Ph.D. degree as a Research Assistant with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis.

From 2005 to 2006, she was a Research Assistant, working on radar waveform design, with the Intelligent Transportation information Systems (ITiS) Laboratory, Tsinghua University. Her research interests lie in the areas of communication theory, and signal processing. Her current research focuses on distributed signal processing for wireless sensor networks and sparsity-exploiting detection with finite-alphabet constraints.

Georgios B. Giannakis (F'97) received the Diploma degree in electrical engineering from the National Technical University of Athens, Greece, in 1981. From 1982 to 1986, he was with the University of Southern California (USC), where 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.

Since 1999, he has been a Professor with the Electrical and Computer Engineering Department at the University of Minnesota, Minneapolis, where he now holds an ADC Chair in Wireless Telecommunications and serves as Director of the Digital Technology Center. His general interests span the areas of communications, networking and statistical signal processing—subjects on which he has published more than 275 journal papers, 450 conference papers, two edited books, and two research monographs titled *Space-Time Coding for Broadband Wireless Communications* (Wiley, 2006) and *Ultra-Wideband Wireless Communications* (Cambridge Press, 2007). Current research focuses on compressive sensing, cognitive radios, network coding, cross-layer designs, mobile ad hoc networks, wireless sensor, and social networks.

G. B. Giannakis is the (co)recipient of seven 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 in 2000, from EURASIP in 2005, a Young Faculty Teaching Award, and the G. W. Taylor Award for Distinguished Research from the University of Minnesota. He is a Fellow of EURASIP and has served the IEEE in a number of posts and also as a Distinguished Lecturer for the IEEE SP Society.

Alfonso Cano (S'01) received the Electrical Engineering degree and Ph.D. degree (Hons.) in telecommunications engineering from the Universidad Carlos III de Madrid, Madrid, Spain, in 2002 and 2006, respectively.

From 2003 to 2006, he was with the Department of Signal Theory and Communications, Universidad Rey Juan Carlos, Madrid, Spain. Since 2007, he has been with the Electrical and Computer Engineering Department at the University of Minnesota, Minneapolis, where he is currently a Post-doctoral Researcher. His research interests lie in the areas of signal processing and communications.