In-Network Channel Decoding Using Consensus on Log-Likelihood Ratio Averages†

Hao Zhu, Alfonso Cano, and Georgios B. Giannakis
Department of Electrical and Computer Engineering, University of Minnesota,
200 Union Street, Minneapolis, MN 55455, USA.

Abstract—This paper considers the problem of distributed decoding of a coded message from a mobile access point (AP) sent to a sensor network. The distributed decoding problem is solved using a consensus algorithm on the log-likelihood ratio averages, which ensures that all sensors attain the same decision as if all received signals were available at a centralized location. Unlike existing distributed hypothesis testing problems, whose inter-sensor communication cost grows exponentially with the codeword size, the novel consensus-based distributed decoder exploits knowledge over the problem to reduce the cost to be linear. Link failures and inter-sensor communication noise are also accounted for. Analysis and simulations confirm that the resultant distributed decoder attains bit error rate performance approaching its centralized counterpart within a few iterations.

Index Terms—Wireless Sensor Networks (WSNs), Distributed Decoding, Consensus Average

I. INTRODUCTION

We consider the problem of cooperatively decoding a message from a mobile access point (AP) by a wireless sensor network (WSN). The goal is to enforce all collaborating sensors to agree on the decoded message. Network agreement is designed so as to enable spatial diversity. Thus, the detection error at the network can be made arbitrarily small by increasing the network size, along with the codeword length. The problem of network agreement is well motivated since sensors may lack of sufficient individual processing to have a reliable message estimation on the received signal. Besides, shadowing and spatial fading may be present, naturally motivating the call for spatial diversity to compensate it.

We seek for optimal local decoding with global information. We gaze this problem deriving messages to be exchanged with neighboring sensors to efficiently describe the information. Messages are derived challenging by efficiency and scalability constraints. As we will show, the optimal maximum-likelihood (ML) distributed block- and bitwise-decoding problem can be reformulated, without loss of optimality, as an equivalent problem described using log-likelihood ratios (LLR). This reformulation naturally leads to a consensus-based distributed algorithm on the LLRs which ensures that each sensor attains the same message as if all received signals were available as a centralized location.

The distributed consensus problem has attracted great interest in the recent years. The distributed consensus averaging problem was investigated in [1], [2], from the perspective of accelerating convergence and minimizing the effects of inter-sensor noise. The consensus estimation problem has been considered by [3] the deterministic signals. The introduction of bridge sensors in [3] exploited the decomposable structure of the centralized optimization problem’s augmented Lagrangian with the method of multipliers [4]. Moreover, for the latter case noise robustness is proved both theoretically and numerically.

The idea of cooperation among nodes for distributed detection problems has also attracted much attention recently. Consensus-based distributed hypothesis testing (DHT) based on the belief consensus algorithm is proposed in [5], and it is shown that this algorithm works for any connected networks with loops. This belief consensus algorithm benefits from scalability, robustness to link failures, and asynchronism in the message sent. Consensus averaging tools are also used in [6] accounting for link failures and noise robustness at the expense of slower convergence rate. From a different perspective, [7] also tackles the problem of DHT using graphical models for the probabilistic model representation. In such scenario, however, convergence may not be guaranteed for any network architecture. Network synchronism is also needed.

Distributed decoding differs from the conventional view at DHT in the sense that decoding also exploits knowledge over the alphabet employed to map the messages at the AP. The novel consensus-based distributed decoder will exploit this fact to reduce communication and computation costs across the network. Specifically, compared to DHT, the novel distributed decoder will update consensus vectors of length considerably smaller than the number of hypothesis.

Notation: Upper (lower) bold face letters will be used for matrices (column vectors); \(1_N\) the \(N \times N\) identity matrix; \(1_N\) the \(N \times 1\) all-one vector; \([\cdot]_{i,j}\) is the \(i, j\)th entry of a matrix; \(|S|\) is the cardinality of the set \(S\); \(\mathcal{CN}(\mu, \sigma^2)\) will stand for a complex Gaussian distribution with mean \(\mu\) and variance \(\sigma^2\). We use \(p(x = z|y)\) to denote the probability density function (pdf) of the random variable (r.v.) \(x\), given the r.v. \(y\), evaluated at \(z\).

II. MODELING & PROBLEM FORMULATION

The problem of distributed decoding by a Cooperative Sensor Network with an AP is illustrated in Fig. 1. The AP broadcasts a common binary codeword \(x\) to an ad hoc WSN.
with $J$ sensors. The codeword $x$ is selected from a linear $(N, K)$ codebook $C$, which is assumed known at both the AP and the sensors. Each codeword $x$ with codeweight $N$ is encoded from the source message $t$ of $K$ independent information bits, but the cardinality of $C$ is $2^K$. For the ad hoc WSN without a central unit (fusion center), the sensor $j$ has to rely only on its local received signal $y_j$, as well as the single-hop communications with neighboring sensors to decode, via “in-network” processing, the information bearing message $x$. Let $N_j \subseteq \{1, \ldots, J\} := J$ denote the set of the sensor $j$’s neighborhood. Inter-sensor links are assumed to be symmetric, and the WSN is modeled as an undirected graph whose vertices are the sensors and its edges represent the available communication links. The graph connectivity information is summarized in the so-called adjacency matrix $A \in \mathbb{R}^{J \times J}$ for which $[A]_{jj} = 1$ if $i \in N_j$, otherwise $[A]_{ji} = 0$. The symmetry of sensor links implies $A = A^T$.

The broadcast channel from the AP to the sensor $j$ is modeled as a discrete memoryless, and is assumed to be independent across sensors. We use the conditional pdf $p[y_j|x]$ to describe the AP-to-sensor channel. We consider the block decoding problem, whereby sensors infer which codeword $x \in C$ or information block $t$ was broadcasted given the received signals $y_1, \ldots, y_J$. The optimal codeword decoder is the solution of the maximum a posteriori (MAP) [8, Section 4.1]:

$$\hat{x}_{ML} = \arg \max_{x \in C} p[x|y_1, \ldots, y_J]. \tag{1}$$

Assuming that messages are equiprobable a priori, the optimal MAP decoder reduces to the maximum likelihood (ML) rule $\hat{x}_{ML} = \arg \max_{x \in C} p[y_1, \ldots, y_J|x]$. By the independence of AP-to-sensor channels, and the memoryless of each of the the AP-to-sensor channels, we can finally write (1) as

$$\hat{x}_{ML} = \arg \max_{x \in C} \sum_{j=1}^{J} \sum_{n=1}^{N} \log p[y_{j,n}|x_n] \tag{2}$$

where $y_{j,n}$ denotes the $n$th entry of $y_j$. Our goal is to develop a distributed decoding algorithm with affordable inter-sensor communications and low computational costs, in order to acquire the centralized optimal decoder (2) in every sensor.

Before continuing, we introduce two more assumptions. These assumptions will be later revisited and thus we name them for later reference:

(a1) The time-invariant WSN communication graph $A$ is connected, i.e., there exists a path connecting any two sensors on the graph;

(a2) The single-hop inter-sensor communication links are ideal.

Similar to [1], [2], [5] and [3], network connectivity in (a1) ensures the utilization of all informative received signal throughout the whole network by the distributed decoding algorithms. The fixed topology in (a1) and the ideal inter-sensor links in (a2) are assumed here for simplicity of discussions and derivations. It will turn out that our distributed decoding algorithm exhibits resilience to communication noise and also adapts to time-varying topologies. Simulations will corroborate this. For further details, we refer the reader to [9].

### III. Consensus-based decoding

The optimization variable $x_n$ in the decoder (2) only takes the value of either 0 or 1, thus, the following equality holds:

$$\log p[y_{j,n}|x_n] = \begin{cases} \log p[y_{j,n}|x_n = 1] & \text{if } x_n = 1 \\ \log p[y_{j,n}|x_n = 0] & \text{if } x_n = 0 \end{cases}$$

$$= x_n \log \left( \frac{p[y_{j,n}|x_n = 1]}{p[y_{j,n}|x_n = 0]} \right) + \log p[y_{j,n}|x_n = 0]. \tag{5}$$

A constant can be subtracted from the cost function in (3) to obtain (see also [10])

$$\hat{x}_{ML} = \arg \min_{x \in C} \sum_{j=1}^{J} \sum_{n=1}^{N} \gamma_{j,n} x_n \tag{4}$$

where $\gamma_{j,n}$ denotes the LLR of each variable $x_n$ at the $j$th sensor, i.e.,

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

Interestingly, as already pointed out in [10] the LLR $\gamma_{j,n}$ can be interpreted by the “cost” of decoding $x_n$ as 1 at the $j$th sensor. In this way, it is very intuitive to see that the centralized ML decoding problem (4) is to find a codeword $\hat{x}_{ML}$ with the minimum “decoding cost”, i.e., the maximum overall likelihood ratio.

Defining the LLR vector at the $j$th sensor $\gamma_j := [\gamma_{j,1}, \ldots, \gamma_{j,N}]$, the optimization problem

$$\hat{x}_{ML} = \arg \min_{x \in C} \sum_{j=1}^{J} \gamma_j^T x = \arg \min_{x \in C} \left( \frac{1}{J} \sum_{j=1}^{J} \gamma_j^T \right) x \tag{6}$$

is equivalent to (4). Notice that in (6), $x$ only depends on the codebook $C$, which is assumed to be available to all $J$ sensors. Thus, in order to solve (6) locally, it suffices for each sensor to acquire the average of LLR vectors

$$\bar{\gamma} = \frac{1}{J} \sum_{j=1}^{J} \gamma_j. \tag{7}$$

![Communication system with broadcasting AP and received sensors.](image)
The average LLR in (7) is a sufficient statistic for the decoding problem in (2). General enough, it can be seen that it also serves as a sufficient statistic for a wide variety of optimal and near-optimal decoding algorithms such as the bitwise decoding algorithm, the Viterbi algorithm for convolutional codes [8, Ch. 7] or iterative decoding of LDPC or concatenated codes [11, Ch 4.1.2.]. See [9] for more details.

IV. Achieving Consensus on LLRs

We have shown in Section III that each sensor can solve the centralized decoding problem using the average LLR $\gamma$. Having the $j$-th sensor an initial vector value $\gamma_j$ we will develop distributed algorithms with only single-hop communications with the neighbors, such that all sensors consents on the average in (7).

We consider two kinds of consensus averaging (CA) algorithms, which achieve consensus on the average of local initial values: i) the weighted-linear CA algorithm ($WL-CA$), which achieves consensus on the average of local initial values; and ii) the method of multipliers CA algorithm ($MoM-CA$), in which the average value is formulated as the solution of an optimization problem using iterations of the method of multipliers (MoM) [4, Sec. 3.4.4] [12, Sec. 4.4].

A. The $WL-CA$ Algorithm

The $WL-CA$ algorithm, has been discussed and examined in plenty of recent works [1], [2], [6], [13]. It is shown that the CA problem can be solved via linear iterations if the weight coefficients are appropriately chosen [1].

The average in (7) is carried component-wise. Thus, for simplicity in exposition, we will derive scalar iterations that corresponds to achieving consensus on the $n$-th entry in (7). The $WL-CA$ algorithm solves this problem with the initialization

$$ r_{j,n}(0) = \gamma_{j,n}, \quad j \in J. $$

Under the assumptions (a1) and (a2), sensor $j$ can perform the following local linear iterations

$$ r_{j,n}(k) = W_{jj} r_{j,n}(k-1) + \sum_{i \in \mathcal{N}_j} W_{ji} r_{i,n}(k-1), \quad j \in J $$

where $k = 1, 2, \ldots$ and $W_{ji}$ is the weight on neighboring sensor's data $r_{i,n}$ at the $j$-th sensor. The consensus on the $n$-th entry of $\gamma$, $\gamma_n$, is achieved as the number of iterations $k$ increases. For simplicity, denoting $r_n(k) := [r_{1,n}(k), \ldots, r_{J,n}(k)]^T$ and setting $W_{ji} = 0$ for $i \notin \mathcal{N}_j$, the iterations in (8) can be stacked in a vector form as

$$ r_n(k) = W r_n(k-1) = W^k r_n(0). $$

To achieve consensus in average, the weight matrix $W$ is chosen such that, for any initial value $r_n(0)$, $r_n(k)$ converges to the average vector $\gamma_n 1_j = (1/J) 1_j^T r(0)$ as $k \to \infty$. The convergence requirements for the weight matrix are discussed in [1].

B. The $MoM-CA$ Algorithm

Section IV-A solves the consensus problem on the average of the LLRs with weighted linear iterations. Here we will show that the CA problem can also been solved from an optimization perspective, with the MoM also based on distributed linear iterations. First, we view the average of LLRs $\gamma_n$ as the minimizer of a quadratic function with consensus constraints (see also [3], [14]):

$$ \{ \hat{s}_{j,n} \}_{j=1}^J = \arg \min_{s_{j,n}} \sum_{j=1}^J (s_{j,n} - \gamma_{j,n})^2 $$

s.t. $s_{j,n} = s_{i,n}$, $j \in J$, $i \in \mathcal{N}_j$. (10)

The constraint in (10) ensures that the auxiliary variable in every sensor reaches consensus, providing a sufficient condition for the optimal solution of and (10) to be

$$ \hat{s}_{j,n} = \gamma_n, \quad j \in J. $$

We can solve (11) by using the MoM with a block coordinate descent algorithm [4, Sec. 3.4.4]. Same strategy utilized in [3] to exploit the decomposition of centralized estimation problem and solve it via distributed local iterations.

Before continuing, let $v_{j,n}^i$ denote the Lagrange multiplier associated with the constraint $s_{j,n} = s_{i,n}$. The augmented Lagrangian for (10) is given by

$$ \mathcal{L}_a(s_{n}, v_{n}) = \sum_{j=1}^J (s_{j,n} - \gamma_{j,n})^2 + \sum_{j=1}^J \sum_{i \in \mathcal{N}_j} v_{j,n}^i (s_{j,n} - s_{i,n}) $$

$$ + \sum_{j=1}^J \sum_{i \in \mathcal{N}_j} c (s_{j,n} - s_{i,n})^2 $$

where $s_{n} = \{ s_{j,n} \}_{j=1}^J$ and $v_{n} = \{ v_{j,n}^i \}_{i \in \mathcal{N}_j}$. The constant $c_n > 0$ is the penalty coefficient corresponding to the constraint $s_{j,n} = s_{i,n}$. While the sensor $j$ maintains the local estimate $s_{j,n}$ and multipliers $\{ v_{j,n}^i \}_{i \in \mathcal{N}_j}$, the MoM is applied to solve the problem (10) with the augmented Lagrangian $\mathcal{L}_a(s_{n}, v_{n})$ in (12) as asserted in the following proposition:

**Proposition 1:** For all $j \in J$, consider iterates $v_{j,n}^i(k)$ and $s_{j,n}(k)$ defined by the recursions

$$ v_{j,n}^i(k+1) = v_{j,n}^i(k) + c_n (s_{j,n}(k) - s_{i,n}(k)), \quad i \in \mathcal{N}_j $$

$$ s_{j,n}(k+1) = \frac{1}{1+2c_n |\mathcal{N}_j|} \left[ \gamma_{j,n} - \sum_{i \in \mathcal{N}_j} v_{j,n}^i(k+1) - v_{i,n}^j(k+1) - 2c_n s_{i,n}(k) \right] $$

(14)

Assuming (a1) and (a2), the iterates $s_{j,n}(k)$ converge to $\gamma_n$ as $k \to \infty$, i.e.,

$$ \lim_{k \to \infty} s_{j,n}(k) = \gamma_n \quad \forall j \in J $$

(15)

for arbitrary initial values $\{ v_{j,n}^i(0) \}_{i \in \mathcal{N}_j}$, $\{ s_{j,n}(0) \}_{j \in J}$ and $c_n < c_{\text{max}}$. 
The proof of Proposition 1 simply follows the iterations in Algorithm 4.4.1 of [12], with the derivations to the augmented Lagrangian (12), so we omit it here for simplicity.

The recursions (13) and (14) constitute the MoM – CA algorithm. Sensor \( j \) maintains the local estimate \( s_{j,n}(k) \) as well as the multipliers \( \{v_{j,n}\}_{n \in N_j} \). During the \((k + 1)\)-st iteration, sensor \( j \) receives the broadcasted estimates \( s_{i,n}(k) \) from all its neighboring sensor \( i \in N_j \), and updates the Lagrangian multiplier via (13), then it transmits the updated multiplier \( v_{j,n}(k + 1) \) to each of its neighboring sensors \( i \in N_j \), based on which each sensor also determines its local estimate \( s_{j,n}(k + 1) \). This completes the \((k + 1)\)-st iteration, after this all the sensors \( j \in J \) broadcast their updated estimates \( s_{j,n}(k + 1) \), which then initialize the next iterations.

We derived two distributed decoding algorithms based on two different consensus averaging algorithms on the LLR vectors. Both of them are guaranteed to converge to the centralized LLR as the iteration number goes to infinity. The decoding problem (2) for sensor \( j \) is performed by substituting the LLR for the \( n \)-th bit by the local estimate of it, either \( r_{j,n}(k) \) or \( s_{j,n}(k) \), depending on which CA algorithm is implemented. Although these two methods are different, and so will be their performance, the iterations in either (9) or (13)–(14) are all simple linear operations for each of the \( N \) variables. The vector that each sensor transmits per iteration has the same length \( N \).

We conclude this section with the following three remarks on our novel distributed decoder.

Remark 1: Another method to derive the distributed iterations to solve the CA problem is to follow the primal-dual algorithm [12, Sec. 4.3], which is based on the Lagrangian of (10) without the quadratic penalty as in (12). [14] introduced this approach and reduced it to the WL – CA algorithm with special initialization. However, the MoM based method, when properly employed and under mild conditions, usually perform better than the primal-dual method. Thus, we expect faster convergence rate of the MoM – CA algorithm than the WL – CA.

Remark 2: The novel consensus-averaging based distributed decoder puts low computation burden on each sensor in that: it does not require each sensor to solve the decoding problem, with exponential complexity in codelength \( N \), in every iteration; moreover, sensors update the consensus vectors of length \( N \) through simple linear iterations. Therefore, the computational complexity per iteration is only \( O(N) \). However, the DHT algorithms reported in [5], [15], [16] need \( O(2^K) \), where \( 2^K \) is the number of possible codes in \( C \). Generally speaking, \( 2^K \) is much less than \( N \) for the binary linear codes in practice.

Remark 3: The consensus-averaging based distributed decoder is performed by the sensor network with low communication costs. First, the \( j \)th sensor only requires the assumed known local channel model \( p[y_j|x] \), and there is no need to flood the information of channel models for every sensor. Second, the length of vectors that the sensor \( j \) transmits per iteration is \( N \), compared to \( 2^K \) in DHT, therefore the consensus-averaging based distributed decoder also puts less communication burden to each sensor.

V. PERFORMANCE ANALYSIS

In this section, we are interested in considering the error performance of our iterative distributed decoding algorithm. We will analyze the performance of the \( WL – CA \) algorithm under additive white Gaussian noise (AWGN) in the AP-to-sensor channel. For the MoM-CA and more general channel models, we refer readers to [9].

For a given linear codebook, the bit error rate (BER) can be upper bounded by a polynomial function of the Bhattacharyya parameter of the channel [8, Section 7.2, 8.2]. For a point-to-point link, the Bhattacharyya parameter is given by [8, Section 6.8]

\[
\Delta := \sum_{y_n \in Y} \sqrt{P[y_n|x_n = 0]}P[y_n|x_n = 1] \quad (16)
\]

where \( Y \) is the set of values \( y_n \) lies in.

We are interested in obtaining bounds on the BER when the algorithm was stopped at a given iteration \( k \). For the AWGN channel with BPSK modulation, the input-output relationship for the channel between AP and the sensor \( j \) can be written as

\[
y_{j,n} = (2x_n - 1) + \eta_{j,n}, \quad x_n = 0, 1 \quad (17)
\]

where the additive noise \( \eta_{j,n} \sim \mathcal{N}(0, \sigma_j^2) \). Interestingly, the LLR for the AWGN channel is a linear transformation of the received signal

\[
\gamma_{j,n} = -\frac{2y_{j,n}}{\sigma_j^2} = -\frac{2 - 4x_n}{\sigma_j^2} - \frac{2\eta_{j,n}}{\sigma_j^2}. \quad (18)
\]

Thus the LLR \( \gamma_{j,n} \) of the AP-to-sensor \( j \) channel given \( x_n \) has the distribution of a Gaussian random variable \( \mathcal{N}((2 - 4x_n)/\sigma_j^2, 4/\sigma_j^2) \). Since the CA algorithms utilized linear iterations on the local LLRs, it turns out that the local estimate of the average LLR \( \gamma_{j,n}(k) := r_{j,n}(k) \) derived by the \( WL – CA \) algorithm at the \( k \)-th iteration will also be a Gaussian random variable given \( x_n \). We can also define an equivalent received signal \( y_{j,n,eq}(k) \) at the \( k \)-th iteration, with pdf \( p[y_{j,n,eq}(k)|x_n] \) Gaussian with mean

\[
m_{j,eq}(k) = \sum_{i \in N_j} |W^k|_{j,i} m_j(0) \quad (19)
\]

where \( m_j(0) = (2 - 4x_n)/\sigma_j^2 \), and variance

\[
\sigma_{j,eq}^2(k) = \sum_{i \in N_j} k_{j,i}(k) \quad (20)
\]

where \( \Sigma(k) = W^k \Sigma(0)(W^T)^k \) and \( \Sigma(k) \) is a diagonal matrix with \( \Sigma_{j,j}(0) = 4/\sigma_j^2 \). With pdf of \( y_{j,n,eq}(k) \) available, \( p[y_{j,n,eq}(k)|x_n] \), we can finally obtain the per-iteration performance bounds plugging it in (16).

Interestingly, the variance \( \sigma_{j,eq}^2(k) \) can be interpreted as the equivalent noise power at the local decoder after the \( k \)-th iteration. This equivalent noise power shrinks at every iteration, converging to \( \lim_{k \to \infty} \sigma_{j,eq}^2(k) = 1/(\sum_{j=1}N_j 1/\sigma_j^2) \), which is exactly the equivalent noise power of the centralized decoder.
VI. SIMULATED EXAMPLES

In this section we test the performance of our distributed decoder with simulated experiments. We implement both \(WL - CA\) and \(MoM - CA\) algorithms. We deploy a WSN with \(J = 10\) sensors. The sensors are randomly placed in the unit square \([0, 1] \times [0, 1]\) with uniform distributed over the 2-dimensional space. The channel between the AP and each sensor is modeled as an AWGN one with BPSK modulation, thus the received noise variance at the sensor \(j\) is set to be \(\sigma_j^2 = 10^{-\rho_j/10}\), where \(\rho_j\) is the corresponding channel signal-to-noise ratio (SNR). For simplicity, we assume that \(\rho_j\) is the same across sensors.

The following three test cases are simulated:

Test Case 1: In this case, each sensor receives a common 2/3 convolutional code, which is encoded at the AP from \(K = 40\) randomly generated information bits, with the resulting codelength \(N = 60\). The distributed decoder is implemented to solve the codeword decoding problem \(2\) using the Viterbi Algorithm \([8]\). \((a1)\) and \((a2)\) are assumed for the inter-sensor communication links and thus the exchange of the data between sensors is considered to be ideal. We implement the consensus iterations \(8\) for the \(WL - CA\) algorithm, and \((13)-(14)\) for the \(MoM - CA\) algorithm, with initial state \(s_{j,n}(0) = \gamma_{j,n}\) for each sensor. In this setup, sensor \(j\)'s local decoder with only its own received information \(y_j\) corresponds to the distributed decoder with iteration number \(k = 0\).

Figure 2 compares the average bit error rate (BER) of the centralized decoder against the distributed one for a variable number of iterations under this scenario in the waterfall region. Clearly, the average BER improves considerably as the number of iterations increases. Specifically, with more iterations the waterfall emerges at the region of much lower SNR and the average BER also falls much more sharply. We can also observe that the \(MoM - CA\) algorithm outperforms the \(WL - CA\) one in that the first one has a smaller error probability with the same number of iterations. Moreover, the waterfall of the novel distributed decoder is very close to that of the centralized one after only 10 iterations, especially for the \(MoM - CA\) algorithm. This is not surprising since the diversity order of the decoder increases as LLR information from neighboring sensors is iteratively incorporated in the average LLR.

Test Case 2: We test here the performance of the bitwise decoding problem, which attempts to obtain the optimal MAP decision for each bit \(t_k\) in \(t\). The MAP decision is made over the a posteriori probability (APP) for each bit, \(P[t_k = a | y_1, \ldots, y_J], a = 0, 1\). The APP is obtained marginalizing over all possible codewords. The bitwise MAP decision thus takes the form:

\[
\hat{t}_k = \arg \max_{a=0,1} P[t_k = a | y_1, \ldots, y_J] = \arg \max_{a=0,1} \sum_{x \in C^t_k} P[x | y_1, \ldots, y_J] \tag{21}
\]

where \(C^t_k := \{x| \text{the } k \text{-th bit of } t \text{'s source message is } a\}\). As demonstrated in [9], the average LLR in \((7)\) is again a sufficient statistic for the decoder in \((21)\).

We test this decoder using the Hamming \((7,4)\) codes to encode a codelength of length \(N = 7\) from the message with \(K = 4\) bits. Similar to test case 1, \((a1)\) and \((a2)\) are assumed, and with the initialization the local decoder corresponds to the distributed decoder with iteration number \(k = 0\). Figure 3 shows a similar behavior of the two algorithms as in test case 1. The distributed decoder based on \(MoM - CA\) algorithm reaches the centralized one’s performance closely within \(k = 10\) iterations, and there is also an observable performance gap between the two CA algorithms.

Test Case 3: Although not discussed in this manuscript, we will relax assumptions \((as1)-(as2)\) and include random link failures between sensors as well as additive noise for both \(WL - CA\) and \(MoM - CA\) algorithms. For an analytical performance claims in this new scenario, we refer readers to [9]. The inter-sensor data exchange is corrupted by additive noise with variance \(\sigma^2\) so that SNR := \(10 \log_{10} E(\gamma T \gamma)/(N \sigma^2) = 20\text{dB}\). Link failures happen with probability 0.1 in every link. The modified \(WL - CA\) follows the one proposed in [6], with the weight matrix at each iteration as \(W(k) = I - \alpha(k) L(k)\), being \(L(k)\) the time-varying graph Laplacian matrix and the stepsize \(\alpha(k) = 1/k\). The modified version of the \(MoM - CA\) algorithm in \((14)-(13)\) is employed whereby estimates and multipliers are kept the same whenever a link failure occurs. Figure 4 compares the average BER of the centralized decoder against the distributed one for a variable number of iterations under this scenario in the waterfall region. Clearly, the average BER improves considerably as the number of iterations increases. Compared to Fig. 2, the drawback of the decreasing stepsize in \(WL - CA\) algorithm emerges as the performance improves slowly after \(k = 10\) iterations. The \(MoM - CA\) algorithm is still very close to the centralized one in around \(k = 10\) iterations, although the gap between the two increases compared to the noise-free case. This is reasonable since the additive noise in the inter-sensor communication links increase the uncertainty of the ML decoding problem.
Fig. 3. Bit-wise Decoder: Average BER versus channel SNR (in dB) for the centralized and distributed decoders with variable number of consensus iterations.

Fig. 4. ML Decoder with link failures and inter-sensor noise: Average BER versus channel SNR (in dB) for the centralized and distributed decoders with variable number of consensus iterations.

VII. CONCLUDING REMARKS

We considered the problem of distributed block- and bitwise-decoding of a coded message from a mobile access point (AP) sent to a large sensor network. Sufficient statistics for decoding were derived resulting in the average over the local log-likelihood ratios. Based on this result, two different distributed methods to compute the average were proposed. One using consensus averaging iterations (WL−CA), another casting the averaging problem as the solution of a distributed optimization problem (MoM−CA). The resulting algorithms feature lower inter-sensor communication costs compared to existing DHT problems. Performance analysis demonstrated high bit error rate performance for a few iterations. Link failures and inter-sensor communication noise are also accounted for in the simulations, demonstrating higher performance by the MoM−CA algorithm.

REFERENCES


