Modeling and Mitigating Errors in Belief Propagation for Distributed Detection

Younes Abdi, *Member, IEEE*, and Tapani Ristaniemi, *Senior Member, IEEE*

***Abstract*— We study the behavior of the belief-propagation (BP) algorithm affected by erroneous data exchange in a wireless sensor network (WSN). The WSN conducts a distributed multi-dimensional hypothesis test over binary random variables. The joint statistical behavior of the sensor observations is modeled by a Markov random field whose parameters are used to build the BP messages exchanged between the sensing nodes. Through linearization of the BP message-update rule, we analyze the behavior of the resulting erroneous decision variables and derive closed-form relationships that describe the impact of stochastic errors on the performance of the BP algorithm. We then develop a decentralized distributed optimization framework to enhance the system performance by mitigating the impact of errors via a distributed linear data-fusion scheme. Finally, we compare the results of the proposed analysis with the existing works and visualize, via computer simulations, the performance gain obtained by the proposed optimization.**

***Index Terms*— Distributed systems, cooperative communications, likelihood-ratio test, communication errors, computation errors, blind signal processing, message-passing algorithms, linear data-fusion, factor graphs.**

I. Introduction

DESIGN of statistical inference systems often involves analysis and modeling of the collective behavior of a group of random variables and their interactions. Consequently, factor graphs, which are commonly used to capture the interdependencies between correlated random variables, provide a powerful framework for developing effective low-complexity inference algorithms in various fields such as wireless communications, image processing, combinatorial optimization, and machine learning, see e.g., [1]–[3]. Belief propagation (BP) [4] is a well-known statistical inference algorithm that works based on parallel message-passing between the nodes in a factor graph. BP is sometimes referred to as the sum-product algorithm.

When working with the BP algorithm, we should bear in mind that digital computation and digital communication are both error-prone processes in general. The messages exchanged between the nodes in a wireless network can always be adversely affected by errors caused by unreliable hardware components, quantization processes, approximate representations, wireless channel impairments, etc. Even though the BP algorithm has been extensively studied in the literature, we have rather limited knowledge about how stochastic errors in messages affect the beliefs obtained and how these erroneous beliefs influence the result of statistical inference schemes implemented by the BP algorithm. This territory is difficult to explore mainly due to the nonlinearities in the BP message-passing iteration.

In [5], we have developed a systematic framework for analyzing the behavior of BP and optimizing its performance in a distributed detection scenario. In particular, we have shown that the decision variables built by the BP algorithm are, approximately, *linear* combinations of the local likelihoods in the network. Consequently, we have derived in [5] closed-form relationships for the system performance metrics and formulated a distributed optimization scheme to achieve a near-optimal detection performance. Moreover, we have discussed the relationship between the BP and the max-product algorithms in [6] where we extend the proposed framework in [5] to optimize the performance of the max-product algorithm in a distributed detection scenario. In this paper, we further extend that framework to gain insight into the impact of computation and communication errors, in a BP iteration, on the resulting decision variables and to effectively mitigate that impact. Examples of BP being used in distributed detection can be found in [7]–[10].

Accumulation of message errors and their adverse effect on the performance of BP is analyzed in [11] where the message errors are modeled as uncorrelated random variables to find probabilistic guarantees on the magnitude of errors affecting the beliefs. The work in [11] is inspired by observing the behavior and stability of digital filters, in the presence of quantization effects, which can be analyzed reliably by assuming uncorrelated behavior in the corresponding random errors [12]. Such a modeling approach is in line with the von Neumann model of noisy circuits [13], which considers transient faults in logic gates and wires as message and node computation noise that is both spatially and temporally independent [14].

The behavior of BP implemented on noisy hardware is investigated in [15] where it is observed that under the so-called *contracting mapping condition* [16], the distance between successive messages in a noise-free BP decreases by the number of iterations. Consequently, in the presence of hardware (or computation) noise, the faulty messages that violate this trend can be detected and discarded (censored) from the BP iterations. Such an approach is termed *censoring*

Manuscript received March 12, 2020; revised September 6, 2020 and January 25, 2021; accepted January 25, 2021. Date of publication February 3, 2021; date of current version May 18, 2021. The associate editor coordinating the review of this article and approving it for publication was A. Cohen. *(Corresponding author: Younes Abdi.)*

Younes Abdi is with the Faculty of Information Technology, University of Jyväskylä, 40014 Jyväskylä, Finland (e-mail: younes.abdi@jyu.fi).

Tapani Ristaniemi, deceased, was with the Faculty of Information Technology, University of Jyväskylä, 40014 Jyväskylä, Finland (e-mail: tapani.ristaniemi@jyu.fi).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCOMM.2021.3056679.

Digital Object Identifier 10.1109/TCOMM.2021.3056679

BP in [15] and is shown to perform well when the hardware noise distribution has a large mass at zero and non-negligible masses at some points sufficiently away from zero. As an alternative approach, the so-called *averaging BP* (ABP) is also proposed in [15]. In this method, as the name implies, an average of the messages up to the last iteration is saved and then used, instead of the actual messages, to build the beliefs. This method is proposed and its convergence is established for general zero-mean computation noise distributions. Again, the von Neumann model is used in [15] to analyze the behavior of message errors.

In this paper, we use the fact that the BP algorithm and the linear data-fusion scheme are elegantly related to each other in the context of distributed detection. Fortunately, there already exists a rich collection of scientific works in the literature that investigate low-complexity detector structures based on linear fusion in various design scenarios [17]–[21]. In many of these works, the data-exchange process within the sensor network is assumed adversely affected by non-idealities in the underlying communication links. Hence, dealing with erroneous data is a familiar challenge faced when designing wireless sensor networks (WSN). We use this knowledge to cope with the impact of message errors on distributed detection systems realized by BP.

It is a common practice to refer to the channels over which the data exchange between the sensing nodes is conducted as *reporting channels* to distinguish them from the channels over which the target signal is detected, which are referred to as *listening channels*. The importance of the present work can be highlighted by noting that distributed detection systems can be highly sensitive to the reporting errors. This phenomenon is illustrated in [22, Sec. IV-C] by a simple example that shows that the overall detection performance cannot go beyond the limits dictated by the reporting channel conditions irrespective of the signal-to-noise-ratio (SNR) levels of the target signal experienced at the listening channels. This effect is further studied and quantified for several hard- and soft-decision fusion schemes in [23]–[25] where it is shown that the reporting channel errors can have a significant impact on the detection performance.

In the existing literature, the reporting channels have been commonly considered nonideal to account for realistic data-exchange processes between the network nodes [17], [20], [21], [26]–[30]. A major application scenario for distributed detection systems is spectrum sensing in cognitive radio networks (CRN) where communication between the sensing nodes is typically conducted without having access to dedicated spectrum bands. This means that the data exchange between those nodes could face stringent constraints in terms of transmit power and bandwidth. Consequently, many works on distributed detection in CRNs focus on reporting links with bandwidth or power constrains, see e.g., [27]. These constraints are typically taken into account, in the system modeling and optimization, by non-ideal reporting links that introduce non-zero bit error probabilities (BEP) in digital [21] and uncorrelated noise in analog reporting schemes [20]. Since the link noise level and BEP are monotonically related to each other [21, Eq. (12)], similar approaches can be used in both analog and digital cases to mitigate the impact of reporting errors, see [17], [21].

In this paper, we are focused on a BP-based decentralized distributed detection scheme. We view message errors in the BP iteration as reporting errors and approximate the messages by a linear expression to study the impact of erroneous data-exchange on the BP algorithm and to clarify how it affects the performance of the resulting distributed detection. We derive approximate expressions that measure the strength of the cumulative errors that affect the BP-based decision variables. These expressions are in the form of mean-squared error (MSE) levels. We compare the MSE levels obtained with the one in [11] to gain insight into the behavior of BP and to see how computation and communication errors propagate throughout the underlying factor graph. Our analysis closely predicts the extent of the deviation of the erroneous decision variables, obtained by an erroneous BP iteration, from their actual values. This is a significant improvement over Ihler's bound in [11]. Moreover, based on the proposed linear approximation, we show that ABP is effective in alleviating message errors and falls short of mitigating the impact of erroneous local likelihood ratios (LLRs) on the resulting decision variables.

We also show, under practical assumptions, that the decision variables built by an erroneous BP are disturbed by a sum of independent error components whose collective impact can be modeled, approximately, by Gaussian random variables. Consequently, we establish the probability distribution of the resulting erroneous decision variables, derive the performance metrics of the BP-based distributed detection in closed form, and propose a two-stage optimal linear fusion scheme to cope with the impact of errors on the system performance. We then develop a *blind* adaptation algorithm to realize the proposed two-stage optimization when the statistics describing the radio environment are not available *a priori*. The proposed blind adaptation modifies the parameters of the BP and the decision threshold at each node, in accordance with the error statistics and channel conditions, to mitigate the impact of errors and to enhance the detection performance. To summarize, we extend the works in [5] and [6] by the following contributions:

- We analyze the behavior of the BP algorithm in the presence of message (and likelihood) errors and derive its performance metrics in closed form in a distributed detection subject to those errors.
- We build a distributed optimization framework for the system that takes into account and effectively mitigates the impact of erroneous data exchange in BP.

Moreover,

- We extend the work in [11] by proposing a tighter error bound that more accurately describes the impact of message errors on the decision variables built by the BP algorithm.
- We extend the work in [15] by analyzing the behavior of ABP. Our work sheds light on ABP's effectiveness and shortcomings.

Here is an overview of the paper organization: In Sec. II, we briefly explain the use of linear fusion and BP in distributed

detection and provide the related formulations. In Sec. III, we discuss errors in BP and model their impact on the decision variables obtained. In Sec. IV, we view BP as a distributed linear fusion and formulate the proposed optimization framework. In Sec. V, we conduct computer simulations to verify our analysis and to illustrate how effectively the proposed method mitigates the impact of errors in a WSN with faulty devices. Finally, we provide our concluding remarks in Sec. VI.

II. Linear Fusion and Belief Propagation for Distributed Detection

We consider N binary random variables, represented by $\boldsymbol{x} = [x_1, \ldots, x_N]^T$, whose status are estimated based on N observations denoted $\boldsymbol{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N]$ made by a network of N sensing nodes. Each node, say node i, which intends to estimate the status of x_i, collects K observation samples, denoted by $\boldsymbol{y}_i = [y_i(1), \ldots, y_i(K)]^T$, and exchanges information with other nodes in the network to realize together a multidimensional hypothesis test as $\hat{\boldsymbol{x}} = \max_{\boldsymbol{x}} p(\boldsymbol{x}|\boldsymbol{Y}) = \max_{\boldsymbol{x}} p(\boldsymbol{Y}|\boldsymbol{x})p(\boldsymbol{x})$. This test can be conducted with low implementation complexity in two alternative ways that are explained in the following.

A. Linear Data-Fusion

Linear fusion has been extensively used in the context of spectrum sensing where the aim is to detect the presence or absence of a target signal by evaluating noisy observations made throughout a WSN. For brevity, we explain the uni-variate case where we have a single binary variable $x \in \{0, 1\}$. The optimal approach to such a detection is known to be the so-called *likelihood-ratio test* (LRT) [31], which is conducted by evaluating the LLR, i.e., by $\hat{x} = \mathbf{1}\{\lambda_{\text{LRT}} - \tau\}$ where

$$\lambda_{\text{LRT}} \triangleq \ln \frac{p(\boldsymbol{Y}|x=1)}{p(\boldsymbol{Y}|x=0)} = \sum_{i=1}^{N} \gamma_i \tag{1}$$

where

$$\gamma_i \triangleq \ln \frac{p(\boldsymbol{y}_i|x=1)}{p(\boldsymbol{y}_i|x=0)} = \boldsymbol{s}_i^T \boldsymbol{y}_i - \frac{1}{2}\|\boldsymbol{s}_i\|^2 \tag{2}$$

where γ_i is referred to as the *local* LLR at node i. By $\mathbf{1}\{\cdot\}$ we represent the indicator function that returns one if its argument is positive and returns zero otherwise. τ is a detection threshold selected via a target false-alarm rate. Eq. (2) indicates that, the LRT is a *matched-filtering* process, which requires the target signal $\boldsymbol{s}_i$ to be known *a priori* at the sensing nodes. Moreover, for Gaussian observations, γ_i in (2) follows a Gaussian distribution. In practice, the local sensing process is realized by *energy detection*, due to its ease of implementation and because its structure does not require the target signal to be known. Energy detection is realized by $\gamma_i \triangleq \frac{1}{K}\|\boldsymbol{y}_i\|^2$ and the sensor outcomes are combined linearly to build a *global* test statistic [17]–[21], i.e.,

$$\lambda_{\text{LF}} \triangleq \sum_{i=1}^{N} w_i \gamma_i = \boldsymbol{w}^T \boldsymbol{\gamma} \tag{3}$$

where $\boldsymbol{w} \triangleq [w_1, \ldots, w_N]^T$ and $\boldsymbol{\gamma} \triangleq [\gamma_1, \ldots, \gamma_N]^T$. Then, λ_{LF} is compared against τ to conduct the hypothesis test, i.e., $\hat{x} = \mathbf{1}\{\lambda_{\text{LF}} - \tau\}$. $\boldsymbol{w}$ can be set to maximize the detection probability. According to the central limit theorem (CLT) [32], when the number of signal samples K is large enough [17]–[21], the outcome of energy detection follows a Gaussian distribution and we can model the test summary λ_{LF}, given the status of x, as a Gaussian random variable. Consequently, the detector performance can be optimized by the well-known *Neyman-Pearson* approach [31]. This optimization is formulated as

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \frac{Q^{-1}(\alpha)\sqrt{\boldsymbol{w}^T\boldsymbol{\Sigma}_0\boldsymbol{w}} - \boldsymbol{w}^T\boldsymbol{\delta}}{\sqrt{\boldsymbol{w}^T\boldsymbol{\Sigma}_1\boldsymbol{w}}} \tag{4}$$

where $\boldsymbol{\delta} \triangleq \boldsymbol{\mu}_1 - \boldsymbol{\mu}_0$ while $\boldsymbol{\mu}_b \triangleq \mathrm{E}[\boldsymbol{\gamma}|x = b]$ and $\boldsymbol{\Sigma}_b \triangleq \mathrm{cov}(\boldsymbol{\gamma}|x = b)$ and $Q^{-1}(\cdot)$ denotes the inverse of the Q-function. α denotes the target false-alarm probability at which the detection probability is maximized. This non-convex problem is solved in [17], [19], [20]. From these works, we know that the performance of linear fusion is close to the LRT performance. Alternatively, we can maximize the so-called *deflection coefficient* of the detector. This approach, which has a low computational complexity and leads to a good performance level, is realized by

$$\boldsymbol{w}^* = \arg\max_{\boldsymbol{w}} \Delta^2(\boldsymbol{w}), \quad \text{s.t.,} \ \|\boldsymbol{w}\| = 1 \tag{5}$$

where

$$\Delta^2(\boldsymbol{w}) \triangleq \frac{\left(\mathrm{E}[\lambda_{\text{LF}}|x=1] - \mathrm{E}[\lambda_{\text{LF}}|x=0]\right)^2}{\mathrm{Var}[\lambda_{\text{LF}}|x=0]} = \frac{\left(\boldsymbol{w}^T\boldsymbol{\delta}\right)^2}{\boldsymbol{w}^T\boldsymbol{\Sigma}_0\boldsymbol{w}} \tag{6}$$

Consequently, by using the Rayleigh-Ritz inequality [17], $\boldsymbol{w}^*$ is obtained in closed form as $\boldsymbol{w}^* = \boldsymbol{\Sigma}_0^{-1}\boldsymbol{\delta} / \left\|\boldsymbol{\Sigma}_0^{-1}\boldsymbol{\delta}\right\|$. Extension of the linear detection structure in (3) to N variables is discussed in [18] in the context of multiband spectrum sensing.

B. Belief Propagation

We model the sensor network structure concerned by an MRF defined on an undirected graph $G = (\mathcal{V}, \mathcal{E})$. In this model, the set of vertices $\mathcal{V}$ corresponds to the set of network nodes while each edge $(i, j) \in \mathcal{E}$ represents a possible connection between nodes i and j. Each node, say node i, is associated with a random variable x_i and the edge (i, j) models a possible correlation between x_i and x_j. This model fits well into the commonly-used ad-hoc network configurations in which major network functionalities are conducted through pairwise i.e., one-hop, links between the nodes located close to each other. This design method is based on the common assumption that nodes located close enough to each other for one-hop communication, experience some levels of correlation between their sensor outcomes.

By using the MRF, we write $p(\boldsymbol{x}|\boldsymbol{Y})$ as a product of univariate and bivariate functions, i.e.,

$$p(\boldsymbol{x}|\boldsymbol{Y}) \propto \prod_{n\in\mathcal{V}} \phi_n(x_n) \prod_{(i,j)\in\mathcal{E}} \psi_{ij}(x_i, x_j) \tag{7}$$

Note that $\propto$ in (7) refers to a normalization that ensures $\sum_{\boldsymbol{x}} p(\boldsymbol{x}|\boldsymbol{Y}) = 1$ and includes but is not limited to $1/p(\boldsymbol{Y})$. When including the bivariate terms in the product, each edge in the factor graph is included in the product only once. This is realized by doing the multiplication on $i < j$ while $i \in \mathcal{N}_j$.

We use $\mathcal{N}_j$ to denote the set of neighbors of node j in the graph, i.e., $\mathcal{N}_j \triangleq \{k : (k,j) \in \mathcal{E}\}$. By using (7), we formulate the message received at node j from node k as

$$\mu_{k\to j}^{(l)}(x_j) \propto \sum_{x_k} \phi_k(x_k)\psi_{kj}(x_k,x_j) \prod_{n\in\mathcal{N}_k^j} \mu_{n\to k}^{(l-1)}(x_k) \quad (8)$$

where by $\mathcal{N}_k^j \triangleq \mathcal{N}_k \backslash \{j\}$ we denote all nodes connected to node k except for node j. We denote by $b_j^{(l)}(x_j)$ the belief, about the status of x_j, formed at node j, which is obtained via multiplying the potential at node j by the messages received from all its neighbors, i.e.,

$$b_j^{(l)}(x_j) \propto \phi_j(x_j) \prod_{k\in\mathcal{N}_j} \mu_{k\to j}^{(l)}(x_j) \quad (9)$$

The beliefs are used as estimates of the desired marginal distributions, i.e., $b_j^{(l)}(x_j) \approx p(x_j|\boldsymbol{Y})$. By adopting the commonly-used exponential model [4] to represent the *a priori* probability measure defined on $\boldsymbol{x}$, we have

$$p(\boldsymbol{x}) \propto \exp\left(\sum_{n\in\mathcal{V}} \theta_n x_n + \sum_{(i,j)\in\mathcal{E}} J_{ij}x_i x_j\right) \quad (10)$$

For notational convenience, we use bipolar binary variables, i.e., $x_j \in \{-1,+1\}$ in our formulations of BP. For a given $\boldsymbol{x}$, we assume the local observations to be mutually independent. Consequently, as explained in [5, Sec. I-B], we have

$$p(\boldsymbol{x}|\boldsymbol{Y}) \propto \prod_{n\in\mathcal{V}} p(\boldsymbol{y}_n|x_n)e^{\theta_n x_n} \prod_{(i,j)\in\mathcal{E}} e^{J_{ij}x_i x_j} \quad (11)$$

Hence, by using (11), the BP messages are built as

$$\mu_{k\to j}^{(l)}(x_j) \propto \sum_{x_k} p(\boldsymbol{y}_k|x_k)e^{\theta_k x_k}e^{J_{kj}x_k x_j} \prod_{n\in\mathcal{N}_k^j} \mu_{n\to k}^{(l-1)}(x_k) \quad (12)$$

and the beliefs at iteration l are expressed as

$$b_j^{(l)}(x_j) \propto p(\boldsymbol{y}_j|x_j)e^{\theta_j x_j} \prod_{k\in\mathcal{N}_j} \mu_{k\to j}^{(l)}(x_j) \quad (13)$$

In the log domain, (12) and (13) convert, respectively, as clarified in Appendix A, to

$$m_{k\to j}^{(l)} = S\left(J_{kj},\ \gamma_k + \sum_{n\in\mathcal{N}_k^j} m_{n\to k}^{(l-1)}\right) \quad (14)$$

$$\lambda_j^{(l)} = \gamma_j + \sum_{k\in\mathcal{N}_j} m_{k\to j}^{(l)} \quad (15)$$

where

$$\lambda_j^{(l)} \triangleq \ln\frac{b_j^{(l)}(x_j=+1)}{b_j^{(l)}(x_j=-1)} \quad (16)$$

$$m_{k\to j}^{(l)} \triangleq \ln\frac{\mu_{k\to j}^{(l)}(x_j=+1)}{\mu_{k\to j}^{(l)}(x_j=-1)} \quad (17)$$

denote, respectively, the estimated likelihood ratio at node j and the message sent to node j from node k while $S(a,b) \triangleq \ln\frac{1+e^{a+b}}{e^a+e^b}$ and $\gamma_k \triangleq \ln\frac{p(\boldsymbol{y}_k|x_k=+1)}{p(\boldsymbol{y}_k|x_k=-1)} = \boldsymbol{s}_k^T\boldsymbol{y}_k - \frac{1}{2}\|\boldsymbol{s}_k\|^2$. In this model, $\boldsymbol{y}_k = \frac{1}{2}(x_k+1)\boldsymbol{s}_k + \boldsymbol{n}_k$ denotes the signal received at node k. Hence, $x_k = -1$ indicates that the target signal $\boldsymbol{s}_k$ is absent leaving the the spectrum free where node k operates. If $x_k = +1$, then the corresponding spectrum band is occupied. J_{kj}'s are calculated as in Eq. (16) in [5] by processing a window of T sensing outcomes. Note that θ_k in (14) is merged into γ_k without having any impact on the rest of the analysis.

After l^* iterations, $\lambda_j^{(l^*)}$ is compared, as a decision variable, against a detection threshold τ_j at node j to decide the status of x_j, i.e., $\hat{x}_j = \mathbf{1}\{\lambda_j^{(l^*)} - \tau_j\}$. By a linear approximation of (14), we have [5]

$$m_{k\to j}^{(l)} \approx c_{jk}\left(\gamma_k + \sum_{n\in\mathcal{N}_k^j} m_{n\to k}^{(l-1)}\right) \quad (18)$$

where $c_{jk} \triangleq \frac{(e^{2J_{kj}}-1)}{(1+e^{J_{kj}})^2}$. This approximation is obtained by the first-order Taylor series expansion, i.e., $S(a,b) \approx S_b(a,0)b$ where $S_b(a,b) = \partial S(a,b)/\partial b$. By using (18) we see that $\lim_{l\to\infty}\lambda_j^{(l)} \approx \lambda_j$ where

$$\lambda_j \triangleq \gamma_j + \sum_{k\in\mathcal{N}_j} c_{jk}\gamma_k + \sum_{k\in\mathcal{N}_j}\sum_{n\in\mathcal{N}_k^j} c_{jk}c_{kn}\gamma_n + \sum_{k\in\mathcal{N}_j}\sum_{n\in\mathcal{N}_k^j}\sum_{m\in\mathcal{N}_n^k} c_{jk}c_{kn}c_{nm}\gamma_m + \dots \quad (19)$$

Therefore, this approximation reveals that, given enough time, all the local likelihood ratios observed in the network are almost linearly combined at node j to calculate its decision variable λ_j. We have shown in [5] that, the convergence of this linear message-passing algorithm is guaranteed when $|c_{j,k}| < \frac{1}{\max_n|\mathcal{N}_n|-1}, \forall(j,k) \in \mathcal{E}$. The linear combination in (19) can be expressed as $\lambda_j = \sum_{i=1}^N a_{ji}\gamma_i$, which is compactly stated in matrix form as

$$\boldsymbol{\lambda} = \boldsymbol{A}\boldsymbol{\gamma} \quad (20)$$

where $\boldsymbol{\lambda} \triangleq [\lambda_1,\dots,\lambda_N]^T$ and $\boldsymbol{A} \triangleq [\boldsymbol{a}_1,\dots,\boldsymbol{a}_N]^T$ while $\boldsymbol{a}_j \triangleq [a_{j1},\dots,a_{jN}]^T$. Here we derive the relationship between $\boldsymbol{A}$ and c_{jk}'s in (19) as

$$\boldsymbol{A} \approx \boldsymbol{I} + \sum_{n=1}^{\infty}\boldsymbol{C}^n - \mathfrak{D}\left(\sum_{n=1}^{\infty}\boldsymbol{C}^n\right) \quad (21)$$

where $\boldsymbol{C} \triangleq [c_{jk}]_{N\times N}$ and $\mathfrak{D}(\boldsymbol{X})$ denotes a diagonal matrix whose main diagonal is equal to that of $\boldsymbol{X}$. The proof is provided in Appendix B.

It is now clear that to have convergence in the message-passing iteration (18), the spectral radius of $\boldsymbol{C}$ has to be less than one. This criterion may be used to impose bounds on c_{jk}'s to guarantee the convergence of the algorithm. Alternatively, the convergence can be guaranteed, without dealing with the complexities of finding the spectral radius, by using the contracting mapping condition as we have discussed in [5]. We use (21) in the following section to derive an estimation of the error strength affecting the decision variables built by an erroneous BP.

III. Errors in Belief Propagation

Eq. (14) shows that at each BP iteration each node creates its messages in terms of its local LLR value as well as the messages received from the neighboring nodes at the previous iteration. In our system model, we assume that the local LLRs and the BP messages are erroneous. As in [11] and [15],

we use the von Neumann approach to modeling the joint statistical behavior of errors.

A. Error Model and Analysis

Since the messages are multiplied together to build the beliefs, we formulate them as multiplicative perturbations affecting true (i.e., error-free) message values, i.e.,

$$\tilde{\mu}_{k\to j}^{(l)}(x_j) = \mu_{k\to j}^{(l)}(x_j)\varepsilon_{k\to j}^{(l)}(x_j) \tag{22}$$

where $\tilde{\mu}_{k\to j}^{(l)}(x_j)$ denotes the erroneous message sent to node j from node k at iteration l while $\varepsilon_{k\to j}^{(l)}(x_j)$ denotes the corresponding error, which is considered in this paper as a stochastic process.

Eq. (22) differs from the model used in [11] in the sense that the error model in that work measures the difference between the messages at iteration l with their counterparts at the fixed point of the message-passing iteration. In other words, the error model in [11] measures the deviation of the messages at each iteration from their final value reached by BP after convergence. The stochastic error we discuss here is briefly studied in [11] under the notion of *additional error*.

By expressing the messages in the the log domain, we have

$$\tilde{m}_{k\to j}^{(l)} \triangleq \ln \frac{\tilde{\mu}_{k\to j}^{(l)}(x_j=+1)}{\tilde{\mu}_{k\to j}^{(l)}(x_j=-1)} = m_{k\to j}^{(l)} + \nu_{k\to j}^{(l)} \tag{23}$$

where

$$\nu_{k\to j}^{(l)} \triangleq \ln \frac{\varepsilon_{k\to j}^{(l)}(x_j=+1)}{\varepsilon_{k\to j}^{(l)}(x_j=-1)} \tag{24}$$

Based on the von Neumann model, we assume that if $k \neq n$, then $E[\ln \varepsilon_{k\to j}^{(l)}(x) \ln \varepsilon_{n\to j}^{(l)}(x)] = 0$ for all x. Consequently, we have $E[\nu_{k\to j}\nu_{n\to j}] = 0$. To measure the collective impact of errors on the belief of node j, we use

$$E_j^{(l)}(x_j) \triangleq \frac{\tilde{b}_j^{(l)}(x_j)}{b_j^{(*)}(x_j)} \tag{25}$$

where $\tilde{b}_j^{(l)}(x_j)$ denotes the belief at node j resulting from a BP iteration with erroneous messages as in (22) while $b_j^{(*)}(x_j)$ denotes the belief of node j at a fixed point reached by an error-free BP iteration. We use $(*)$ instead of (l) to indicate the messages and beliefs at a fixed point of the error-free BP.

By assuming uncorrelated stochastic behavior for the message errors, an upper bound on cumulative errors affecting the beliefs can be obtained. Specifically, assuming $\mathrm{Var}\left[\nu_{k\to j}^{(l)}\right] \leq (\ln u)^2$ for all k, j, l, an upper bound on the resulting cumulative strength of errors at node j is derived in [11] as,

$$\mathrm{E}\left[\left\{\ln d\left(E_j^{(l)}\right)\right\}^2\right] \leq \sum_{k\in\mathcal{N}_j}\left(\sigma_{kj}^{(l)}\right)^2 \tag{26}$$

where $\sigma_{kj}^{(1)} = \ln d(\psi_{kj})^2$ and

$$\left(\sigma_{kj}^{(l+1)}\right)^2 = \left(\ln \frac{d(\psi_{kj})^2\omega_{kj}^{(l)}+1}{d(\psi_{kj})^2+\omega_{kj}^{(l)}}\right)^2 + (\ln u)^2 \tag{27}$$

while

$$\left(\ln \omega_{kj}^{(l)}\right)^2 = \sum_{n\in\mathcal{N}_k^j}\left(\sigma_{nk}^{(l)}\right)^2 \tag{28}$$

where

$$d\left(E_j^{(l)}\right) \triangleq \sup_{a,b}\sqrt{\frac{E_j^{(l)}(a)}{E_j^{(l)}(b)}} \tag{29}$$

$$d(\psi_{kj})^2 \triangleq \sup_{a,b,c,d}\frac{\psi_{kj}(a,b)}{\psi_{kj}(c,d)} \tag{30}$$

We use the upper bound in (26) in the log domain based on the fact that (see (16) and (25))

$$\tilde{\lambda}_j^{(l)} \triangleq \ln \frac{\tilde{b}_j^{(l)}(+1)}{\tilde{b}_j^{(l)}(-1)} = \lambda_j^{(*)} + \ln \frac{E_j^{(l)}(+1)}{E_j^{(l)}(-1)} \tag{31}$$

which leads to

$$\begin{aligned}\mathrm{E}\left[\left|\tilde{\lambda}_j^{(l)} - \lambda_j^{(*)}\right|^2\right] &= \mathrm{E}\left[\left|\ln E_j^{(l)}(+1) - \ln E_j^{(l)}(-1)\right|^2\right] \\ &= \mathrm{E}\left[\left\{\ln d\left(E_j^{(l)}\right)\right\}^2\right] \leq \sum_{k\in\mathcal{N}_j}\left(\sigma_{kj}^{(l)}\right)^2\end{aligned} \tag{32}$$

Hence, in the detection structure discussed, (26) gives an upper bound on the MSE level observed in the decision variable at node j.

B. Linear Approximations

In our analysis, we distinguish between the message errors and the errors in the computation of local LLRs to gain further insight into the behavior of the BP algorithm. In particular, we model the erroneous local LLRs as $\tilde{\gamma}_k \triangleq \gamma_k + \epsilon_k$ and refer to ϵ_k's as *likelihood errors* (LE) while assuming that LEs are uncorrelated as well, i.e., $E[\epsilon_k\epsilon_n] = 0$ for $k \neq n$. We refer to $\nu_{k\to j}$'s as *message errors* (ME) and assume that LEs and MEs are mutually independent. Moreover, we assume that all MEs and LEs are independent of the messages and of the local LLRs. Note that *the bound in* (32) *does not take LEs into account.*

Taking both types of error into account, we express the messages decision variables as

$$\tilde{m}_{k\to j}^{(l)} = S\left(J_{kj},\ \tilde{\gamma}_k + \sum_{n\in\mathcal{N}_k^j}\tilde{m}_{n\to k}^{(l-1)}\right) + \nu_{k\to j}^{(l)} \tag{33}$$

$$\tilde{\lambda}_j^{(l)} = \tilde{\gamma}_j + \sum_{k\in\mathcal{N}_j}\tilde{m}_{k\to j}^{(l)} \tag{34}$$

which shows that the errors pass through the same nonlinear transformation (i.e., S) as the messages do. By using (33), we can analyze the behavior of errors. The proposed linear BP iteration in the presence of message errors is expressed as

$$\tilde{m}_{k\to j}^{(l)} \approx c_{jk}\left(\tilde{\gamma}_k + \sum_{n\in\mathcal{N}_k^j}\tilde{m}_{n\to k}^{(l-1)}\right) + \nu_{k\to j}^{(l)} \tag{35}$$

Consequently, similar to the way (19) is derived, the resulting erroneous decision variable is formed as

$$\begin{aligned}\tilde{\lambda}_j^{(l)} \approx \Big[\tilde{\gamma}_j + \sum_{k\in\mathcal{N}_j}c_{jk}\tilde{\gamma}_k + \sum_{k\in\mathcal{N}_j}\sum_{n\in\mathcal{N}_k^j}c_{jk}c_{kn}\tilde{\gamma}_n + \dots\Big] \\ + \sum_{k\in\mathcal{N}_j}\nu_{k\to j}^{(l)}\end{aligned} \tag{36}$$

which can be reorganized as

$$\tilde{\lambda}_j^{(l)} \approx \lambda_j + \xi_j^{(l)} \tag{37}$$

where

$$\xi_j^{(l)} \triangleq \sum_{i=1}^{N} a_{ji}\epsilon_i + \sum_{k\in\mathcal{N}_j} \nu_{k\to j}^{(l)} \tag{38}$$

Eq. (38) shows that the error affecting the decision variable at node j has two distinct components. The first component is built as a linear combination of LEs while the second one is the sum of the MEs received at node j from its one-hop neighbors. The first component is fixed whereas the second one exhibits a new realization at every iteration.

According to (38), deviation from the error-free decision variables, caused by errors in the BP iterations, can approximately be measured by

$$\mathrm{E}\left[\left|\tilde{\lambda}_j^{(l)} - \lambda_j^{(*)}\right|^2\right] \approx \mathrm{E}\left[\left|\xi_j^{(l)}\right|^2\right] = \boldsymbol{a}_j^T \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} \boldsymbol{a}_j + \mathrm{tr}\left(\boldsymbol{\Sigma}_{\boldsymbol{\nu}_j}\right) \tag{39}$$

where $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} \triangleq \mathrm{cov}(\boldsymbol{\epsilon})$ and $\boldsymbol{\Sigma}_{\boldsymbol{\nu}_j} \triangleq \mathrm{cov}\left(\boldsymbol{\nu}_j^{(l)}\right)$ while $\boldsymbol{\epsilon} \triangleq [\epsilon_1, \ldots, \epsilon_N]^T$ and $\boldsymbol{\nu}_j^{(l)}$ denotes an $|\mathcal{M}_j|$-by-1 vector that contains $\nu_{k\to j}^{(l)}$'s for $k \in \mathcal{M}_j$ where $\mathcal{M}_j \triangleq \mathcal{N}_j \cup \{j\}$ while $\nu_{j\to j}^{(l)} \triangleq 0$. Note that (36) includes more ME terms than just $\sum_{k\in\mathcal{N}_j} \nu_{k\to j}^{(l)}$. However, they can all be neglected since $|c_{jk}| < 1$ for all j, k.

Eq. (36) shows that when BP is used to realize a distributed detection, the erroneous local likelihoods in the network are combined linearly to build the decision variables. We can evaluate the impact of the errors on the system performance by analyzing the stochastic behavior of the erroneous decision variables $\tilde{\lambda}_j^{(l)}$. Given $\boldsymbol{x}$, the decision variable at node j is obtained as a linear combination of independent random variables. Consequently, its conditional pdf is derived as

$$f_{\tilde{\lambda}_j|\boldsymbol{x}}(z|\boldsymbol{b}) \approx \overset{N}{\underset{i=1}{\circledast}} \frac{1}{a_{ji}} f_{\tilde{\gamma}_i|\boldsymbol{x}}\left(\frac{z}{a_{ji}}|\boldsymbol{b}\right) * \underset{k\in\mathcal{N}_j}{\circledast} f_{\nu_{k\to j}}(z) \tag{40}$$

where

$$f_{\tilde{\gamma}_i|\boldsymbol{x}}(z|\boldsymbol{b}) = f_{\gamma_i|\boldsymbol{x}}(z|\boldsymbol{b}) * f_{\epsilon_i}(z) \tag{41}$$

while $\circledast$ and $*$ denote the convolution operator. Consequently, we have

$$\begin{aligned} g_j(\tau_j, v) &\triangleq \Pr\{\tilde{\lambda}_j > \tau_j | x_j = v\} \\ &= \sum_{\boldsymbol{b}\in\{-1,1\}^{N-1}} p_{\boldsymbol{x}_{(j)}|x_j}(\boldsymbol{b}|v) \int_{\tau_j}^{\infty} f_{\tilde{\lambda}_j|\boldsymbol{x}}\left(z|\mathcal{E}_{j,v}(\boldsymbol{b})\right) dz \end{aligned} \tag{42}$$

where $v \in \{-1, +1\}$, $\boldsymbol{x}_{(j)} \triangleq [x_1, x_2, \ldots, x_{j-1}, x_{j+1}, \ldots, x_N]^T$ and $\mathcal{E}_{j,v}(\boldsymbol{b}) \triangleq \{\boldsymbol{x}_{(j)} = \boldsymbol{b}, x_j = v\}$ while $p_{\boldsymbol{x}_{(j)}|x_j}(\boldsymbol{b}|v) \triangleq \Pr\{\boldsymbol{x}_{(j)} = \boldsymbol{b}|x_j = v\}$. Solving $g_j(\tau_j, -1) = \alpha$ gives a threshold value that fixes the false-alarm rate at α. Similarly, $g_j(\tau_j, 1) = \beta$ fixes the detection rate at β. Recall that a_{ji}'s are found by using c_{jk}'s, see (21).

As a common practical case, when the local LLRs and the errors follow Gaussian distributions [17]–[21] the decision variable $\tilde{\lambda}_j$ follows a Gaussian distribution as well and it is fully characterized by its first- and second-order statistics. Specifically, we have

$$\int_{\tau_j}^{\infty} f_{\tilde{\lambda}_j|\boldsymbol{x}}\left(z|\mathcal{E}_{j,v}(\boldsymbol{b})\right) dz = Q\left(\frac{\tau_j - \mu_{j,v}(\boldsymbol{b})}{\sigma_{j,v}(\boldsymbol{b})}\right) \tag{43}$$

where

$$\begin{aligned} \mu_{j,v}(\boldsymbol{b}) &\triangleq \mathrm{E}\left[\tilde{\lambda}_j|\mathcal{E}_{j,v}(\boldsymbol{b})\right] \\ &= \mathrm{E}\left[\gamma_j|x_j = v\right] + \sum_{i\neq j} a_{ji}\mathrm{E}\left[\gamma_i|x_i = b_i\right] \end{aligned} \tag{44}$$

$$\begin{aligned} \sigma_{j,v}^2(\boldsymbol{b}) &\triangleq \mathrm{Var}\left[\tilde{\lambda}_j|\mathcal{E}_{j,v}(\boldsymbol{b})\right] \\ &= \mathrm{Var}\left[\gamma_j|x_j = v\right] + \sum_{i\neq j} a_{ji}^2 \mathrm{Var}\left[\gamma_i|x_i = b_i\right] + \mathrm{E}\left[|\xi_j|^2\right] \end{aligned} \tag{45}$$

In (44) we have assumed, without loss of generality, zero-mean errors. Note that, without the proposed approximation these performance measures are not available analytically due to the nonlinearity of (14). In the rest of the paper, we assume that the local likelihoods, LEs, and MEs are Gaussian random variables. Eq. (40) shows that, according to the CLT, even if the local LLRs and errors are not Gaussian random variables, the stochastic behavior of the decision variables can still be approximately described by Gaussian distributions.

C. Impact of Averaging

In ABP, the message-passing iteration is the same as in BP. However, instead of the actual message values, an average of the messages are used to build the decision variables. To be more specific, in the log domain and for $l \geq L + 1$, let

$$\bar{m}_{k\to j}^{(l)} \triangleq \frac{1}{L+1} \sum_{t=l-L}^{l} \tilde{m}_{k\to j}^{(t)} \tag{46}$$

The decision variable at node j is calculated by

$$\bar{\lambda}_j^{(l)} \triangleq \gamma_j + \sum_{k\in\mathcal{N}_j} \bar{m}_{k\to j}^{(l)} \tag{47}$$

Similar to our discussion regarding (19), we can show that when the message-passing iteration is error-free, $\bar{\lambda}_j^{(*)} \triangleq \lim_{l\to\infty} \bar{\lambda}_j^{(l)} = \lambda_j$. Hence, we can see that the averaging process does not alter the fixed points achieved by the error-free linear BP. This observation is in line with the convergence analysis provided in [15].

The impact of averaging on LEs and MEs can be clarified by noting that

$$\bar{\lambda}_j^{(l)} = \lambda_j + \bar{\xi}_j^{(l)} \tag{48}$$

where, assuming L to be large enough, we have

$$\bar{\xi}_j^{(l)} = \sum_{i=1}^{N} a_{ji}\epsilon_i + \sum_{k\in\mathcal{N}_j} \bar{\nu}_{k\to j}^{(l)} \approx \sum_{i=1}^{N} a_{ji}\epsilon_i \tag{49}$$

since $\bar{\nu}_{k\to j}^{(l)} \triangleq \frac{1}{L+1}\sum_{t=l-L}^{l} \nu_{k\to j}^{(t)} \approx 0$. We can state (49) in the form of MSE as

$$\mathrm{E}\left[\left|\bar{\lambda}_j^{(l)} - \lambda_j^{(*)}\right|^2\right] \approx \boldsymbol{a}_j^T \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} \boldsymbol{a}_j + \frac{1}{L+1}\mathrm{tr}\left(\boldsymbol{\Sigma}_{\boldsymbol{\nu}_j}\right) \tag{50}$$

Assuming L to be large enough and MEs to have zero mean, (49) shows that the resulting decision variable built by

ABP in (47) is almost cleared of MEs. However, the averaging process has almost no impact on LEs.

Note that in ABP the message-passing iteration is the same as in BP and the averaging is only performed when computing the decision variables. Moreover, in ABP, instead of storing the messages in past iterations separately, we only need to store the sum of the messages up to the current iteration. As a consequence, the number of additional memory cells required can be kept constant [15]. We will use ABP in Sec. IV-B to build an offline learning-optimization structure for the linear BP in the presence of errors.

IV. Mitigating Errors by Linear Fusion

In this section, we first propose a two-stage linear fusion scheme to obtain a near-optimal detection performance by suppressing the impact of the errors. Then, we realize the proposed optimization in a blind decentralized setting where the required statistics are not available a priori.

A. Linear Fusion

First, since $|c_{jk}| < 1$, we further approximate the decision variable λ_j in (19) as

$$\lambda_j \approx \sum_{k\in\mathcal{M}_j} c_{jk}\gamma_k \tag{51}$$

Due to the symmetry of the data-fusion process in (19), the approximation in (51) is an effective approach to building a distributed computing framework for system performance optimization. In this framework, each node interacts only with its immediate neighbors. We have clarified this symmetry in [5, Sec. III-B]. By taking into account the errors while analyzing the linear BP, (19) and (36) lead to

$$\tilde{\lambda}_j \approx \sum_{k\in\mathcal{M}_j} c_{jk}\left(\gamma_k+\epsilon_k\right) + \sum_{k\in\mathcal{N}_j} \nu_{k\to j} \tag{52}$$

We see that the disturbance on the decision variable caused by LEs is built, approximately, as a linear combination of ϵ_k's with c_{jk}'s acting as weights in this combination. Therefore, we use c_{jk}'s as design parameters to mitigate the impact of ϵ_k's. Moreover, MEs are combined in (52) linearly and in this combination, all weights are one. We propose to extend this combination by using a modified version of (34) as

$$\hat{\lambda}_j^{(l)} \triangleq \tilde{\gamma}_j + \sum_{k\in\mathcal{N}_j} w_{jk}\tilde{m}_{k\to j}^{(l)} \tag{53}$$

This modification in the structure of the decision variable does not affect the convergence of the proposed linear BP since it does not alter the message-passing iteration. Now, based on an approximation similar to the one in (52), we have

$$\hat{\lambda}_j \approx \sum_{k\in\mathcal{M}_j} w_{jk}c_{jk}\left(\gamma_k+\epsilon_k\right) + \sum_{k\in\mathcal{N}_j} w_{jk}\nu_{k\to j} \tag{54}$$

Since $\hat{\lambda}_j$ is a Gaussian random variable, we only need its mean and variance to characterize its statistical behavior. Specifically, for $b\in\{-1,1\}$, we have

$$\Pr\{\hat{\lambda}_j > \tau_j | x_j = b\} = Q\left(\frac{\tau_j - \mathrm{E}[\hat{\lambda}_j|x_j=b]}{\mathrm{Var}[\hat{\lambda}_j|x_j=b]}\right) \tag{55}$$

where

$$\mathrm{E}[\hat{\lambda}_j|x_j=b] \approx \boldsymbol{v}_j^T\boldsymbol{\mu}_b \tag{56}$$

$$\mathrm{Var}[\hat{\lambda}_j|x_j=b] \approx \boldsymbol{v}_j^T\left(\boldsymbol{\Sigma}_{\boldsymbol{\gamma}_j|b}+\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}_j}\right)\boldsymbol{v}_j + \boldsymbol{w}_j^T\boldsymbol{\Sigma}_{\boldsymbol{\nu}_j}\boldsymbol{w}_j \tag{57}$$

where $\boldsymbol{v}_j \triangleq \boldsymbol{w}_j\circ\boldsymbol{c}_j$ in which $\circ$ denotes the Hadamard product while $\boldsymbol{\Sigma}_{\boldsymbol{\gamma}_j|b} = \mathrm{cov}(\boldsymbol{\gamma}_j|x_j=b)$ and $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}_j} = \mathrm{cov}(\boldsymbol{\epsilon}_j)$. Moreover, $\boldsymbol{w}_j$, $\boldsymbol{c}_j$, $\boldsymbol{\gamma}_j$, and $\boldsymbol{\epsilon}_j$ are $|\mathcal{M}_j|$-by-1 vectors containing w_{ji}'s, c_{ji}'s, γ_i's, and ϵ_i's for $i\in\mathcal{M}_j$, respectively. Eq. (55) gives the system false-alarm probability for $b=-1$ and the detection probability for $b=1$. The false-alarm probability can be set to $P_{\mathrm{f}}^{(j)} = \alpha$ by

$$\tau_j = Q^{-1}(\alpha)\mathrm{Var}[\hat{\lambda}_j|x_j=-1] + \mathrm{E}[\hat{\lambda}_j|x_j=-1] \tag{58}$$

and then by using (55) – (57), $\boldsymbol{w}_j$ and $\boldsymbol{c}_j$ can jointly be optimized in a Neyman-Pearson setting.

To avoid the challenges of this optimization, we maximize the deflection coefficient of the detector. We already know that the resulting detector performs well when the decision variables follow Gaussian distributions. In this manner, we mitigate the joint impact of LEs and MEs with low computational complexity.

The proposed optimization is conducted in two consecutive stages based on the fact that we can decompose the construction of $\hat{\lambda}_j$ into two consecutive fusion processes. That is, we first optimize c_{jk}'s by considering the impact of ϵ_k's on γ_k's. Then, we consider the resulting scaled LLRs, i.e., $c_{jk}\gamma_k$'s, as new statistics to be linearly combined, while being weighted by w_{jk}'s and distorted by $\nu_{k\to j}$'s, to make the decision variable at node j.

More specifically, first, we optimize $\boldsymbol{c}_j$ in a hypothetical linear detector with its decision variable defined as

$$\hat{\lambda}_j' \triangleq \boldsymbol{c}_j^T\left(\boldsymbol{\gamma}_j+\boldsymbol{\epsilon}_j\right) \tag{59}$$

The coefficients resulting from this optimization scale up the more reliable local LLRs, with respect to the ones built under low SNR regimes, to suppress the effect of LEs. We denote the resulting fusion weights by $\boldsymbol{c}_j^*$. Then, we use $\boldsymbol{c}_j^*$ within the structure of the actual detector to optimize $\boldsymbol{w}_j$ to mitigate the impact of MEs. That is, we consider the following linear detector at node j

$$\hat{\lambda}_j'' \triangleq \boldsymbol{w}_j^T\left(\boldsymbol{\chi}_j+\boldsymbol{\nu}_j\right) \tag{60}$$

where $\boldsymbol{\chi}_j \triangleq \boldsymbol{c}_j^*\circ(\boldsymbol{\gamma}_j+\boldsymbol{\epsilon}_j)$ contains χ_{jk}'s for $k\in\mathcal{M}_j$ while $\chi_{jk} = c_{jk}^*(\gamma_k+\epsilon_k)$. The vector $\boldsymbol{\nu}_j$ contains $\nu_{k\to j}$'s with $k\in\mathcal{M}_j$. In this structure, the elements of χ_{jk}'s are seen as the actual local LLRs that are combined to build the decision variable at node j while the combination takes into account the joint degrading effect of MEs and LEs.

Based on the material provided in Sec. II-A, the first stage of the proposed optimization is formally stated as

$$\boldsymbol{c}_j^* = \arg\max_{\boldsymbol{c}_j} \Delta_j'(\boldsymbol{c}_j),\ \text{s.t.},\ \|\boldsymbol{c}_j\| = 1 \tag{61}$$

where

$$\Delta_j'(\boldsymbol{c}_j) = \frac{\left(\boldsymbol{c}_j^T\boldsymbol{\delta}_j\right)^2}{\boldsymbol{c}_j^T\left(\boldsymbol{\Sigma}_{\boldsymbol{\gamma}_j|-1}+\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}_j}\right)\boldsymbol{c}_j} \tag{62}$$

where $\boldsymbol{\delta}_j \triangleq \mathrm{E}[\boldsymbol{\gamma}_j|x_j=1] - \mathrm{E}[\boldsymbol{\gamma}_j|x_j=-1]$. The resulting $\boldsymbol{c}_j^*$ is then used to realize the second stage of the proposed

optimization by solving

$$\boldsymbol{w}_j^* = \arg\max_{\boldsymbol{w}_j} \Delta_j''(\boldsymbol{w}_j), \text{ s.t., } \|\boldsymbol{w}_j\| = 1 \tag{63}$$

where

$$\Delta_j''(\boldsymbol{w}_j) = \frac{\left(\boldsymbol{w}_j^T \hat{\boldsymbol{\delta}}_j\right)^2}{\boldsymbol{w}_j^T \left(\boldsymbol{\Sigma}_{\boldsymbol{\chi}_j|-1} + \boldsymbol{\Sigma}_{\boldsymbol{\nu}_j}\right) \boldsymbol{w}_j} \tag{64}$$

where $\hat{\boldsymbol{\delta}}_j = \boldsymbol{c}_j^* \circ \boldsymbol{\delta}_j$ and $\boldsymbol{\Sigma}_{\boldsymbol{\chi}_j|-1} = \text{cov}(\boldsymbol{\chi}_j|x_j = -1) = \boldsymbol{c}_j^* \boldsymbol{c}_j^{*T} \circ \left(\boldsymbol{\Sigma}_{\boldsymbol{\gamma}_j|-1} + \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}_j}\right)$. Having $\boldsymbol{c}_j^*$ and $\boldsymbol{w}_j^*$, the detection threshold τ_j is derived as $\tau_j = Q^{-1}(\alpha)\text{Var}[\lambda_j''|x_j = -1] + \text{E}[\lambda_j''|x_j = -1]$ to fix the system false-alarm rate at α.

The convergence condition $|c_{j,k}| < \frac{1}{\max_n |\mathcal{N}_n|-1}, \forall (j,k) \in \mathcal{E}$ can be realized by a simple normalization of $c_{j,k}^*$'s since the objective function in (61) does not change by normalizing its argument.

Through the proposed two-stage optimization, we enhance the detection performance at node j by suppressing the joint impact of MEs and LEs with low computational complexity. The statistics required in this optimization are collected from the one-hop neighbors of node j. This makes the proposed method a viable approach in ad-hoc network configurations where major network functionalities are conducted through one-hop links between the network nodes.

Algorithm 1 Blind Adaptation of Fusion Weights in Erroneous Linear Belief Propagation

Input: $\tilde{\boldsymbol{\gamma}}_T$, $\bar{\boldsymbol{\gamma}}_T$,$\boldsymbol{\tau}^{(0)}$, $\kappa_{\max}$, η
Output: Near-optimal $\boldsymbol{c}_j$ and $\boldsymbol{w}_j$ for $j = 1, \ldots, N$

1. Let $\kappa \leftarrow 0$ and initialize $\hat{\boldsymbol{x}}^{(0)}$ by comparing $\tilde{\boldsymbol{\gamma}}_T$ against $\boldsymbol{\tau}^{(0)}$;
2. **while** $\kappa \leq \kappa_{\max}$
3. **for** node $j \in \{1, 2, \ldots, N\}$
4. Calculate $E[\bar{\gamma}_i|\hat{x}_j^{(\kappa)}]$ and $\text{cov}(\bar{\gamma}_i, \bar{\gamma}_k|\hat{x}_j^{(\kappa)})$ for all $i, k \in \mathcal{M}_j$;
5. Solve (61) to find $\boldsymbol{c}_j^{(\kappa)}$ and $\tau_j^{(\kappa)}$;
6. Set $\boldsymbol{c}_j^*$ by an η-test on $\boldsymbol{c}_j^{(\kappa)}$;
7. **end**
8. Use $\boldsymbol{c}_j^*$'s and $\tau_j^{(\kappa)}$'s to run linear ABP on $\tilde{\boldsymbol{\gamma}}_T$ to find $\hat{\boldsymbol{x}}^{(\kappa+1)}$;
9. $\kappa \leftarrow \kappa + 1$;
10. **end**
11. **for** node $j \in \{1, 2, \ldots, N\}$
12. Use $\boldsymbol{c}_j^*$ and $\hat{\boldsymbol{x}}^{(\kappa_{\max})}$ to calculate $\hat{\boldsymbol{\delta}}_j$, $\boldsymbol{\Sigma}_{\boldsymbol{\chi}_j|0}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\nu}_j}$;
13. Solve (63) to find $\boldsymbol{w}_j^*$;
14. **end**
15. Output $\boldsymbol{c}_j^*$ and $\boldsymbol{w}_j^*$ for $j = 1, 2, \ldots N$;

B. Offline Learning and Adaptation

To realize the proposed optimization, we need the mean and covariance of the local erroneous LLRs. In a blind setting where there is no prior information available regarding the radio environment, we have to estimate those parameters based on the detection outcomes. The main challenge here is that the state of x_j is required at node j while the only information available in practice is the detection outcome $\hat{x}_j$. Hence, node j has to estimate the conditional statistics required in (61) and (63) based on $\hat{x}_j$. The problem with such an adaptation mechanism is that it makes the detection outcome $\hat{x}_j$ depend on those estimates. This dependence creates an inherent deteriorating loop by feeding the detection errors back into the system structure through erroneous estimates of the required statistics.

To overcome this challenge, we propose an extended version of the blind learning-adaptation loop in [5] that accommodates the proposed error-mitigating structure. The pseudo-code of this adaptation is provided in Algorithm 1 where the task of each node is specified in a distributed computing framework. Algorithm 1 operates on a window of stored sensing outcomes and involves a secondary BP that is run much less frequently than the rate at which the distributed detection is performed. The outcomes of this offline BP are used in the estimation of the required unknown statistics. In this adaptation, the desired optimizations are realized iteratively while each node interacts only with its one-hop neighbors. Consequently, Algorithm 1 can be well incorporated in a decentralized network configuration.

In the sequel, we propose a blind adaptation structure in which we use κ to denote the iteration index. Note that we use l as the iteration index in the main BP through which the distributed detection is realized. The offline adaptation updates the fusion weights in the proposed linear BP by processing T stored samples of $\tilde{\boldsymbol{\gamma}}$. This window of erroneous local likelihoods is denoted by $\tilde{\boldsymbol{\gamma}}_T$ and contains samples of $\tilde{\boldsymbol{\gamma}}(t)$ for $t = 1, 2, \ldots, T$. Recall that, $\tilde{\boldsymbol{\gamma}} = \boldsymbol{\gamma} + \boldsymbol{\epsilon}$ where $\boldsymbol{\epsilon}$ denotes the vector of LEs. The offline detection outcomes at iteration κ are denoted by $\hat{\boldsymbol{x}}^{(\kappa)} \triangleq [\hat{x}_1^{(\kappa)}, \ldots, \hat{x}_N^{(\kappa)}]$ while the resulting fusion weights and detection thresholds are denoted $\boldsymbol{c}_j^{(\kappa)}$ and $\tau_j^{(\kappa)}$ respectively. $\hat{\boldsymbol{x}}^{(\kappa)}$ denotes a window of stored sensing outcomes $\hat{\boldsymbol{x}}^{(\kappa)}(t)$ for $t = 1, 2, \ldots, T$. For simplicity, we do not show the time index when dealing with $\tilde{\boldsymbol{\gamma}}_T$, and $\hat{\boldsymbol{x}}^{(\kappa)}$.

Due to errors caused by the wireless links between the sensing nodes, node j does not have access to $\tilde{\gamma}_k(t)$, $k \in \mathcal{N}_j$. Specifically, what node j receives from node k is $\tilde{\gamma}_k(t) + \nu_{k \to j}$ where $\nu_{k \to j}$ denotes the corresponding link error. Without loss of generality, we attribute MEs to wireless link errors. To alleviate the link errors, before starting the adaptation process node j receives L copies of $\tilde{\gamma}_k(t)$ from node k and calculates an average to obtain $\bar{\gamma}_k(t) \triangleq \tilde{\gamma}_k(t) + \bar{\nu}_{k \to j}$ where $\bar{\nu}_{k \to j}$ denotes the average of L independent realizations of $\nu_{k \to j}$. The desired statistics are then calculated by processing $\bar{\gamma}_k$'s, which approximate $\tilde{\gamma}_k$'s. We use $\bar{\boldsymbol{\gamma}}_T$ to contain the samples of $\bar{\gamma}_k(t)$ for $t = 1, 2, \ldots, T$ for $k = 1, 2, \ldots, N$.

In a realistic detection scenario, the data exchanged between the nodes in the proposed offline adaptation is impaired by both types of errors. Since in the first linear fusion (61) we take into account the impact of LEs only, we need to isolate this optimization from the MEs. To this end, we estimate the desired statistics by using linear ABP. As we saw in Sec. III-C, MEs do not affect the ABP outcomes significantly. Therefore, the resulting offline decision variables are almost cleared of

MEs and closely realize (61). Note that, the offline linear ABP processes $\tilde{\bar{\gamma}}_T$ (not $\bar{\gamma}_T$) since each node, say node j, builds its own messages by using its own local likelihood $\tilde{\gamma}_j$. The outcomes of the linear ABP are then used in processing $\bar{\gamma}_T$ as indicated in line 4 of Algorithm 1.

As indicated in line 5, based on the outcomes of the ABP, we suppress the impact of LEs by the first-stage linear fusion. The resulting fusion coefficients enhance the quality of the linear ABP, which, in turn, enhances the quality of the fusion coefficients obtained in the following iteration. By repeating this learning-optimization cycle, we suppress the impact of LEs significantly while this cancellation process is not disturbed by the MEs. See lines 2 – 10 in Algorithm 1.

In this section, we distinguish between the coefficients obtained by Algorithm I and the ones obtained by linearizing the BP algorithm as in (18). Specifically, we use $\mathbf{c}_j^{\text{BP}}$ to collect c_{jk}^{BP}'s for $k \in \mathcal{M}_j$ while $c_{jk}^{\text{BP}} \triangleq (e^{2J_{kj}}-1)/(1+e^{J_{kj}})^2$. Line 6 indicates what we refer to as the η-*test* that ensures the system performance level not to fall below that of the legacy BP algorithm. The test is as follows: given a predefined value η and for $n = 1, 2, \ldots, N$, if $\boldsymbol{c}_j^{\text{BP}}(n)/\boldsymbol{c}_j^{(\kappa)}(n) \geq \eta$ then $\boldsymbol{c}_j^*(n) \leftarrow \boldsymbol{c}_j^{\text{BP}}(n)$. Otherwise, $\boldsymbol{c}_j^*(n) \leftarrow \boldsymbol{c}_j^{(\kappa)}(n)$. That is, we do not use a coefficient obtained by the offline optimization if that coefficient is not large enough with respect to its corresponding coefficient in the main linearized BP. The reason is that, when the primary-user signal received at node j is buried under heavy noise, the local optimization at node j is not able to fully capture the correlations between x_j and its neighbors. Consequently, the resulting c_{jk}'s attain values too small to maintain a good detection performance. In such cases, we replace those coefficients with their counterparts in the linearized BP. This technique prevents the resulting coefficients from degrading the performance when the corresponding nodes operate under an SNR regime that is not good enough for a reliable estimation of the desired statistics. We use the legacy BP coefficients for those nodes.

Having the high-quality fusion weights $\boldsymbol{c}_j^*$ and more-reliable detection outcomes $\hat{\boldsymbol{x}}^{(\kappa_{\max})}$, we now realize the second stage of the proposed optimization. To this end, node j finds $\text{cov}(\boldsymbol{\chi}_j|x_j=-1)$ in (64) by calculating $\text{cov}(\bar{\gamma}_j|\hat{x}_j^{(\kappa_{\max})}=-1)$ and then multiplying the result, element-wise, by $\boldsymbol{c}_j^*\boldsymbol{c}_j^{*T}$. Note that $\text{cov}(\bar{\gamma}_j|x_j) \approx \boldsymbol{\Sigma}_{\boldsymbol{\gamma}_j|x_j} + \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}_j}$ where $\bar{\gamma}_j$ denotes the jth column in $\bar{\gamma}_T$. The elements of the diagonal matrix $\boldsymbol{\Sigma}_{\boldsymbol{\nu}_j}$ are found at node j by noting that $\bar{\gamma}_k(t) \approx \tilde{\gamma}_k(t)$, which indicates that $\text{Var}[\nu_{k\to j}] \approx \text{Var}[\tilde{\gamma}_k(t)+\nu_{k\to j}] - \text{Var}[\bar{\gamma}_k(t)]$. Lines 11 – 14 in Algorithm 1 indicate the second stage of the proposed optimization.

In case a certain performance level, such as a certain false-alarm rate, is required to be guaranteed, we can use the *detector calibration* technique in [5]. The thresholds obtained by Algorithm I appear to be too sensitive to errors in the estimated statistics. Therefore, we do not use Algorithm I for threshold adaptation. Note that the implementation complexity of the BP algorithm does not increase significantly by using Algorithm I since the channel statistics change slowly compared to the rate at which the spectrum sensing is conducted. In other words, Algorithm I is executed far less frequently than is the spectrum sensing. Note that the spectrum sensing is performed at every time slot.

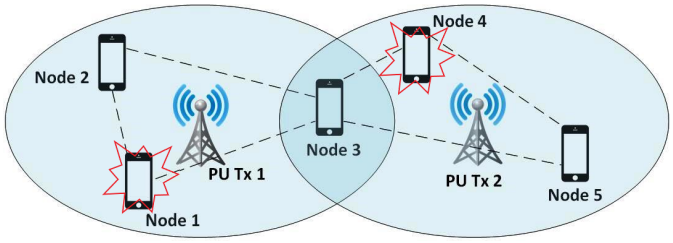

Fig. 1. We have five secondary users cooperating via BP to sense the radio spectrum allocated to a primary network with two transmitters. We use dashed lines to depict the links between the cooperating nodes, through which the BP messages are exchanged. Nodes 1 and 4 act as faulty nodes in the second experiment.

V. Numerical Results

Our simulation scenario in this section is an extension of the one considered in [5]. We consider a spectrum sensing scenario typically used in CRNs [33]. Specifically, we have five sensing nodes, as secondary users, cooperating with each other via BP to find spectral opportunities not in use by the primary users. Fig. 1 depicts our network configuration where the range of primary transmitter 1 covers nodes 1, 2, and 3 while the range of primary transmitter 2 covers nodes 3, 4, and 5. We use dashed lines to represent the links between the cooperating nodes.

Each node generates its local sensing outcome by using energy detection while processing 100 samples of the received signals. Node 1 and node 5 receive the primary-user signal with an SNR level of -5 dB, node 2 and node 4 experience an SNR level of -8 dB in the received primary-user signal, and node 3 receives signals from both of the primary transmitters at -10 dB each. In our simulations, we randomly switch the primary transmitters on and off. We realize these on-off periods by generating correlated binary random variables. Hence, the primary transmitters exhibit correlated random behavior in our simulations. This is an extension to the primary network behavior assumed in [7]. In that work, one of the primary transmitters is on while the other one is off and they do not change their status. As in [5] and [7], we assume that the channel coefficients are fixed during a time slot.

We conduct two experiments in this section. In the first experiment, whose results are depicted in Fig. 2, we evaluate our analysis and compare its results to the one obtained by Ihler *et. al.* in [11]. In particular, we compare the levels of what we define as the *decision SNR* (DSNR), predicted by our analysis, against the one predicted by the work in [11]. We define the DSNR level at node j as

$$\rho_{\text{D}}^{(j)} \triangleq \frac{E\left[|\lambda_j|^2\right]}{E\left[|\tilde{\lambda}_j - \lambda_j|^2\right]} \tag{65}$$

This parameter measures the ratio of the power of the decision variable built by an error-free BP to the power of the error affecting the same decision variable in an erroneous BP.

We realize the LEs and MEs as uncorrelated zero-mean Gaussian random variables in our simulations. At each node,

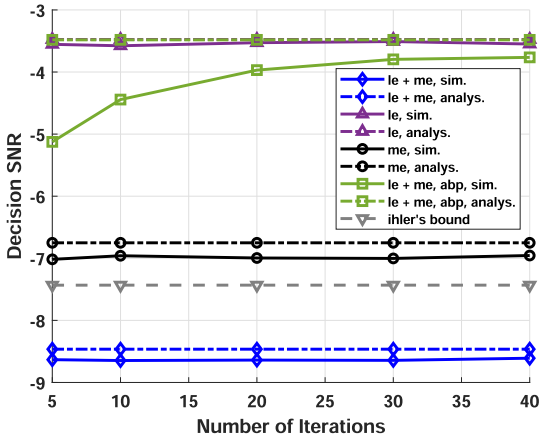

Fig. 2. Impact of errors on the decision variables built by the BP and ABP algorithms.

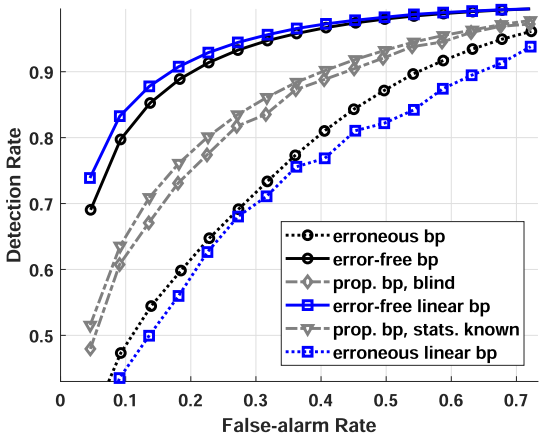

Fig. 3. Performance levels of BP, linear BP, and the proposed optimal linear BP in the presence of errors.

we measure the strength of LEs and MEs with respect to the node's local likelihood. Specifically, we define the *LE SNR* level at node j as

$$\rho_{\text{LE}}^{(j)} \triangleq \frac{E\left[|\gamma_j|^2\right]}{E\left[|\tilde{\gamma}_j - \gamma_j|^2\right]} \tag{66}$$

while we define the *ME SNR* level at node j and for $i \in \mathcal{N}_j$ as

$$\rho_{\text{ME}}^{(j,i)} \triangleq \frac{E\left[|\gamma_j|^2\right]}{E\left[|\tilde{m}_{j\to i} - m_{j\to i}|^2\right]} \tag{67}$$

Consequently, now we have a reference level at each node to measure the power of LEs and MEs injected by that node into the BP iteration.

To see the impact of different error types separately as well as together, we run three BP algorithms in the first experiment. The first one is affected only by LEs, the second one is affected only by MEs and the third one is affected by both of the error types concurrently. Moreover, we evaluate the behavior of ABP in the presence of both error types. In each case, the average of the DSNR level predicted by the proposed analysis and the one observed in simulations are depicted in Fig. 2. The average of the DSNR is calculated over all of the sensing nodes in the network. The dashed curves in Fig. 2 represent the results of our analysis while the solid curves show the average DSNR levels observed in simulations. In this experiment, we consider $\rho_{\text{LE}}^{(j)} = \rho_{\text{ME}}^{(j,i)} = 10$ dB for all i, j and $\zeta = 1$. In Fig. 2, for each data point, we have averaged 20,000 realizations of the decision variables. For the adaptation process in Algorithm I, we have used 2,000 data samples. That is, $T = 2000$ in this experiment.

In Fig. 2, we see a close match between the results predicted by our analysis and the ones obtained via simulations. We also see that our analysis provides a better estimate of the DSNR levels than Ihler's bound in [11]. *Our analysis provides a better estimation even when we only consider MEs.*

Moreover, we see a gap in Fig. 2 between the DSNR levels of the LE-only and ME-only cases. This gap indicates that the impact of MEs is more deteriorating than the impact of LEs when we have the same levels of ρ_{LE} and ρ_{ME}. This appears to be a reasonable observation in our network since the set of error items injected to the BP iteration by each node, say node j, comprises of only one LE and $|\mathcal{N}_j|$ MEs and $|\mathcal{N}_j| \geq 1$. This observation can be justified based on the linearity of the proposed detection scheme. Specifically, the average number of neighbors in our network is $(4 \times 2 + 4)/5 = 2.4$ and $10\log_{10}(2.4) \approx 3.8$ dB and we see almost 3.5 dB gap between the two curves. Note that Ihler's bound (26) cannot distinguish between the LEs and MEs.

Fig. 2 confirms our observation in Sec. III-C, where we showed that ABP is quite resilient to the impact of MEs. We now see that, by increasing the number of iterations the average DSNR level of ABP approaches that of a BP that is affected by LEs only. Note that the ABP is affected by both types of errors and even when the number of iterations is rather low, the DSNR level of ABP is quite high compared to that of a regular BP affected by both error types. This observation justifies our choice of ABP for the offline learning and optimization cycle in Algorithm 1.

In this experiment, for a given number of iterations N_{iterate}, we set L large enough, i.e., $L \geq N_{\text{iterate}}$, to use all the messages generated by BP when realizing the averaging process in the ABP decision variable (46). Hence, by increasing the number of iterations, L is increased and this increase leads to a heavier suppression of MEs. We can clearly see in Fig. 2 the increase in the DSNR level of ABP caused by increasing N_{iterate}. Moreover, the DSNR level of the ABP approaches that of the LE-only case, which, as we predicted in Sec. III-C, indicates that the LEs are not affected by the averaging process in the ABP.

In addition, our analysis, which is based on the von Neumann model, predicts that the DSNR levels of an erroneous BP algorithm do not change by the number of iterations significantly. This is also confirmed by the simulation results in Fig. 2.

In the second experiment, we study the impact of errors on the detection performance of the sensor network depicted in Fig. 1. We assume that nodes 1 and 4 are faulty and inject errors into the BP algorithm. All other nodes operate in a reliable manner, meaning that the errors in their local likelihoods and messages are negligible. The results are depicted in Fig. 3 where for each data point we have averaged 100,000 detection results. For the adaptation of the linear BP, we have used a

window of 2,500 detection outcomes. That is $T = 2500$ in Algorithm I. We have set $L = 10$ in the offline ABP of Algorithm 1. In the faulty nodes we have $\rho_{\text{LE}}^{(1)} = \rho_{\text{LE}}^{(4)} = 10$ dB while $\rho_{\text{ME}}^{(1)} = \rho_{\text{ME}}^{(4)} = 20$ dB. The aim of this experiment is to see whether the proposed method is able to alleviate the impact of those faulty nodes on the overall detection performance.

As for performance metrics, we use the average of the detection and false-alarm rates observed in all of the sensing nodes. We consider both the BP and linear BP algorithms with error-free and erroneous iterations. Consequently, we see how each detection method is affected by errors. Moreover, we consider the proposed linear BP optimized with and without having the required statistics. Fig. 3 shows that, in the presence of LEs and MEs, the detection performance of both message-passing algorithms are significantly degraded. This observation clarifies the need for a better BP algorithm, which resists the impact of errors. In Fig. 3, we also see that the proposed method significantly improves the detection rate of the system in the presence of errors. Moreover, we can see that the proposed blind adaptation scheme closely achieves the optimal performance level when the required statistics are not available a priori.

VI. Conclusion

We studied the impact of computation and communication errors on the behavior of the BP algorithm. We showed that when evaluating the impact of errors on a distributed detection conducted by BP, the detection can effectively be modeled as a distributed linear data-fusion scheme. Consequently, we can analyze its statistical behavior in the presence of errors and obtain closed-form relations for its performance metrics. Moreover, by optimizing the resulting linear data-fusion we can effectively suppress the impact of errors and obtain a better detection performance.

Appendix A
Notes on Eq. (14)

By inserting (12) into (17) and then dividing both the numerator and denominator in (17) by the factor $p(\boldsymbol{y}_k|-1)e^{-\theta_k - J_{kj}} \prod_{n\in\mathcal{N}_k^j} \mu_{n\to k}^{(l-1)}(-1)$, we obtain $m_{k\to j}^{(l)} = S\left(2J_{kj},\ 2\theta_k + \gamma_k + \sum_{n\in\mathcal{N}_k^j} m_{n\to k}^{(l-1)}\right)$. Now we see that, in the proposed linearized BP, θ_k only imposes a shift on the value of the decision variable λ_j. This shift has no impact on the detection process since the detection threshold is a degree of freedom, i.e., an optimization variable, in our design. In other words, we merge $2\theta_k$ into the detection threshold τ_j. Moreover, since J_{kj} is also a degree of freedom, we can simply use J_{kj} instead of $2J_{kj}$. Recall that we jointly optimize J_{kj}'s and the detection thresholds.

Appendix B
Proof of Eq. (21)

We focus on non-diagonal elements of $\boldsymbol{A}$ here since it is clear that the diagonal ones are all close to one. It is straightforward to see that (21) holds for $l = 1, 2, 3$. That is, we have $\boldsymbol{\lambda}^{(l)} = \boldsymbol{A}^{(l)}\boldsymbol{\gamma}$ where $\boldsymbol{A}^{(l)} \approx \sum_{n=1}^{l} \boldsymbol{C}^n$ for $l \le 3$. Based on this observation, we prove (21) by induction. Specifically, we show that if $\boldsymbol{A}^{(k)} \approx \sum_{n=1}^{k} \boldsymbol{C}^n$ for $k \le l$ then $\boldsymbol{A}^{(l+1)} \approx \sum_{n=1}^{l+1} \boldsymbol{C}^n$ or, equivalently, $\boldsymbol{A}^{(l+1)} \approx \boldsymbol{A}^{(l)} + \boldsymbol{C}^{l+1}$. From (15) and (18) and by setting $c_{jk} = 0$ for $k \notin \mathcal{N}_j, \forall j$ we have

$$\left[\boldsymbol{A}^{(l+1)}\right]_{ji} = \left[\boldsymbol{A}^{(l)}\right]_{ji} + \sum_{k_1\neq j}\sum_{k_2\neq j}\cdots \sum_{k_{l-1}\neq k_{l-3}}\sum_{k_l\neq k_{l-2}} c_{jk_1}c_{k_1k_2}\ldots c_{k_{l-1}k_l}c_{k_li} \quad (68)$$

Note that $\left[\boldsymbol{A}^{(l)}\right]_{ji} = \partial\lambda_j/\partial\gamma_i$. Hence, to prove (21) we need to show that

$$\left[\boldsymbol{C}^{l+1}\right]_{ji} \approx \sum_{k_1\neq j}\sum_{k_2\neq j}\cdots\sum_{k_l\neq k_{l-2}} c_{jk_1}c_{k_1k_2}\ldots c_{k_li} \quad (69)$$

According to (68), our induction hypothesis indicates for $n \le l$ that

$$[\boldsymbol{C}^n]_{ji} \approx \sum_{k_1\neq j}\sum_{k_2\neq j}\cdots\sum_{k_{n-1}\neq k_{n-3}} c_{jk_1}c_{k_1k_2}\ldots c_{k_{n-1}i} \quad (70)$$

To show that (69) is true, we rewrite its right-hand side (RHS) as

$$\begin{aligned}\text{RHS} = &\sum_{k_l=1}^{N} c_{k_li}\sum_{k_1\neq j}\sum_{k_2\neq j}\cdots\sum_{k_{l-1}\neq k_{l-3}} c_{jk_1}c_{k_1k_2}\ldots c_{k_{l-1}k_l}\\ &-\sum_{k_1\neq j}\sum_{k_2\neq j}\cdots\sum_{k_{l-1}\neq k_{l-3}} c_{jk_1}c_{k_1k_2}\\ &\ldots c_{k_{l-2}k_{l-1}}c_{k_{l-1}k_{l-2}}c_{k_{l-2}i}\end{aligned} \quad (71)$$

which, based on (70), means that

$$\text{RHS} \approx \sum_{k_l=1}^{N} c_{k_li}\left[\boldsymbol{C}^l\right]_{jk_l} - O_j^{(l+1)} \quad (72)$$

where $O_j^{(l+1)}$ denotes the right-most term in (71).

Now, we recognize the outcome of the first sum in (72) as $\left[\boldsymbol{C}^{l+1}\right]_{ji}$ which is what we were looking for. Since the induction hypothesis states that $\boldsymbol{A}^{(l)}$ contains $\boldsymbol{C}^{l-1}$ as one of its summands, the offset term $O_j^{(l+1)}$ can be neglected. The reason is that

$$\left|\frac{O_j^{(l+1)}}{\left[\boldsymbol{C}^{l-1}\right]_{ji}}\right| \le \left|\left(|\mathcal{N}_{k_{l-1}}| - 1\right)\tilde{c}^2\right| \quad (73)$$

where $\tilde{c} \triangleq \frac{1}{\max_n |\mathcal{N}_n|-1} < 1$ and, therefore, $|\tilde{c}|^2 \ll 1$. The bound in (73) is derived by replacing the term $c_{k_{l-2}k_{l-1}}c_{k_{l-1}k_{l-2}}$ within $O_j^{(l+1)}$ by $\tilde{c}^2$. Recall that, to ensure the system convergence we have $|c_{j,k}| < \tilde{c}, \forall(j,k) \in \mathcal{E}$. Hence, the proof is complete.

Acknowledgment

The authors would like to warmly thank the Editor and the anonymous reviewers for their valuable comments.

References

[1] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[2] N. Noorshams and M. J. Wainwright, "Stochastic belief propagation: A low-complexity alternative to the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 59, no. 4, pp. 1981–2000, Apr. 2013.

[3] H. Loeliger, "An introduction to factor graphs," *IEEE Signal Process. Mag.*, vol. 21, no. 1, pp. 28–41, Jan. 2004.
[4] M. J. Wainwright and M. I. Jordan, "Graphical models, exponential families, and variational inference," *Found. Trends Mach. Learn.*, vol. 1, nos. 1–2, pp. 1–305, 2007.
[5] Y. Abdi and T. Ristaniemi, "Optimization of linearized belief propagation for distributed detection," *IEEE Trans. Commun.*, vol. 68, no. 2, pp. 959–973, Feb. 2020.
[6] Y. Abdi and T. Ristaniemi, "The max-product algorithm viewed as linear data-fusion: A distributed detection scenario," *IEEE Trans. Wireless Commun.*, vol. 19, no. 11, pp. 7585–7597, Nov. 2020.
[7] F. Penna and R. Garello, "Decentralized Neyman-Pearson test with belief propagation for peer-to-peer collaborative spectrum sensing," *IEEE Trans. Wireless Commun.*, vol. 11, no. 5, pp. 1881–1891, May 2012.
[8] H. Wymeersch, F. Penna, and V. Savic, "Uniformly reweighted belief propagation for estimation and detection in wireless networks," *IEEE Trans. Wireless Commun.*, vol. 11, no. 4, pp. 1587–1595, Apr. 2012.
[9] H. Li, "Cooperative spectrum sensing via belief propagation in spectrum-heterogeneous cognitive radio systems," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Apr. 2010, pp. 1–6.
[10] Z. Zhang, Z. Han, H. Li, D. Yang, and C. Pei, "Belief propagation based cooperative compressed spectrum sensing in wideband cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 10, no. 9, pp. 3020–3031, Sep. 2011.
[11] A. T. Ihler, W. F. John, III, and A. S. Willsky, "Loopy belief propagation: Convergence and effects of message errors," *J. Mach. Learn. Res.*, vol. 6, no. May, pp. 905–936, 2005.
[12] A. S. Willsky, "Relationships between digital signal processing and control and estimation theory," *Proc. IEEE*, vol. 66, no. 9, pp. 996–1017, Sep. 1978.
[13] J. von Neumann, "Probabilistic logic and the synthesis of reliable organisms from unreliable components," in *Automata Studies*. Princeton, NJ, USA: Princeton Univ. Press, 1956, pp. 43–98.
[14] L. R. Varshney, "Performance of LDPC codes under faulty iterative decoding," *IEEE Trans. Inf. Theory*, vol. 57, no. 7, pp. 4427–4444, Jul. 2011.
[15] C.-H. Huang, Y. Li, and L. Dolecek, "Belief propagation algorithms on noisy hardware," *IEEE Trans. Commun.*, vol. 63, no. 1, pp. 11–24, Jan. 2015.
[16] J. M. Mooij and H. J. Kappen, "Sufficient conditions for convergence of the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4422–4437, Dec. 2007.
[17] Z. Quan, S. Cui, and A. H. Sayed, "Optimal linear cooperation for spectrum sensing in cognitive radio networks," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 1, pp. 28–40, Feb. 2008.
[18] Z. Quan, S. Cui, A. H. Sayed, and H. V. Poor, "Optimal multiband joint detection for spectrum sensing in cognitive radio networks," *IEEE Trans. Signal Process.*, vol. 57, no. 3, pp. 1128–1140, Mar. 2009.
[19] Z. Quan, W.-K. Ma, S. Cui, and A. H. Sayed, "Optimal linear fusion for distributed detection via semidefinite programming," *IEEE Trans. Signal Process.*, vol. 58, no. 4, pp. 2431–2436, Apr. 2010.
[20] G. Taricco, "Optimization of linear cooperative spectrum sensing for cognitive radio networks," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 1, pp. 77–86, Feb. 2011.
[21] Y. Abdi and T. Ristaniemi, "Joint local quantization and linear cooperation in spectrum sensing for cognitive radio networks," *IEEE Trans. Signal Process.*, vol. 62, no. 17, pp. 4349–4362, Sep. 2014.
[22] K. B. Letaief and W. Zhang, "Cooperative communications for cognitive radio networks," *Proc. IEEE*, vol. 97, no. 5, pp. 878–893, May 2009.
[23] S. Chaudhari, J. Lunden, and V. Koivunen, "BEP walls for collaborative spectrum sensing," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2011, pp. 2984–2987.
[24] S. Chaudhari, J. Lundén, V. Koivunen, and H. V. Poor, "BEP walls for cooperative sensing in cognitive radios using K-out-of-N fusion rules," *Signal Process.*, vol. 93, no. 7, pp. 1900–1908, Jul. 2013.
[25] S. Chaudhari, J. Lunden, V. Koivunen, and H. V. Poor, "Cooperative sensing with imperfect reporting channels: Hard decisions or soft decisions?" *IEEE Trans. Signal Process.*, vol. 60, no. 1, pp. 18–28, Jan. 2012.
[26] W. Zhang and K. Letaief, "Cooperative spectrum sensing with transmit and relay diversity in cognitive radio networks–[transaction letters]," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 4761–4766, Dec. 2008.
[27] C. Sun, W. Zhang, and K. Letaief, "Cooperative spectrum sensing for cognitive radios under bandwidth constraints," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2007, pp. 1–5.
[28] C. Sun, W. Zhang, and K. B. Letaief, "Cluster-based cooperative spectrum sensing in cognitive radio systems," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2007, pp. 2511–2515.
[29] S.-H. Lee, D.-C. Oh, and Y.-H. Lee, "Hard decision combining-based cooperative spectrum sensing in cognitive radio systems," in *Proc. Int. Conf. Wireless Commun. Mobile Comput. Connecting World Wirelessly (IWCMC)*, New York, NY, USA, 2009, pp. 906–910, doi: 10.1145/1582379.1582576.
[30] B. Chen and P. Willett, "On the optimality of the likelihood-ratio test for local sensor decision rules in the presence of nonideal channels," *IEEE Trans. Inf. Theory*, vol. 51, no. 2, pp. 693–699, Feb. 2005.
[31] S. M. Kay, *Fundamentals of Statistical Signal Processing*. Upper Saddle River, NJ, USA: Prentice-Hall, 1993.
[32] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*. New York, NY, USA: McGraw-Hill, 2002.
[33] I. F. Akyildiz, B. F. Lo, and R. Balakrishnan, "Cooperative spectrum sensing in cognitive radio networks: A survey," *Phys. Commun.*, vol. 4, no. 1, pp. 40–62, Mar. 2011.

Younes Abdi (Member, IEEE) received the B.Sc. degree in electrical engineering from the University of Tabriz, Tabriz, Iran, in 2008, the M.Sc. degree in electrical engineering from Tarbiat Modares University, Tehran, Iran, in 2011, and the Ph.D. degree in information technology from the University of Jyväskylä, Jyväskylä, Finland, in 2016.

From 2010 to 2011, he was with the Radio Communications Group, Iran Telecommunications Research Center, Tehran, where he was involved in the standardization and regulatory issues of cognitive radio networks. Since 2012, he has been with the Faculty of Information Technology, University of Jyväskylä, working on distributed detection systems. From 2013 to 2016, he served as a member of Working Group 1900.1 in the IEEE Dynamic Spectrum Access Networks Standards Committee. He was a recipient of research grants from the Finnish National Graduate School in Electronics, Telecommunications, and Automation and Jyväskylä Doctoral Program in Computing and Mathematical Sciences. During spring and summer 2018, he was a Visiting Researcher with Nokia Bell Labs, Espoo, Finland, where he was a member of the Radio Working Group in the MulteFire Alliance. He is currently with Nokia Solutions and Networks, Espoo, where he is working on 5G technology.

Tapani Ristaniemi (Senior Member, IEEE) received the M.Sc. degree in mathematics, the Ph. Lic. degree in applied mathematics, and the Ph.D. degree in wireless communications from the University of Jyväskylä, Jyväskylä, Finland, in 1995, 1997, and 2000, respectively. In 2001, he was a Professor with the Department of Mathematical Information Technology, University of Jyväskylä. In 2004, he was with the Department of Communications Engineering, Tampere University of Technology, Tampere, Finland, where he was appointed as a Professor of wireless communications. In 2006, he moved back to the University of Jyväskylä to take up his appointment as a Professor of computer science. In 2013, he was a Visiting Professor with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He was also an Adjunct Professor with the Tampere University of Technology. He has authored or coauthored over 150 publications in journals, conference proceedings, and invited sessions. He served as a Guest Editor of IEEE WIRELESS COMMUNICATIONS in 2011. He was an Editorial Board Member of *Wireless Networks* and *International Journal of Communication Systems*.