

Convergence Analysis of Gaussian Belief Propagation for Distributed State Estimation

Tianju Sui, Damian E. Marelli and Minyue Fu

Abstract—Belief propagation (BP) is a well-celebrated iterative optimization algorithm in statistical learning over network graphs with vast applications in many scientific and engineering fields. This paper studies a fundamental property of this algorithm, namely, its convergence behaviour. Our study is conducted through the problem of distributed state estimation for a networked linear system with additive Gaussian noises, using the weighted least-squares criterion. The corresponding BP algorithm is known as Gaussian BP. Our main contribution is to show that Gaussian BP is guaranteed to converge, under a mild regularity condition. Our result significantly generalizes previous known results on BP's convergence properties, as our study allows general network graphs with cycles and network nodes with random vectors. This result is expected to inspire further investigation of BP and wider applications of BP in distributed estimation and control.

I. INTRODUCTION

Pearl's Belief Propagation, or *Belief Propagation* (BP) for short, is a well-celebrated algorithm in the area of statistical learning. Originally proposed by Pearl [1] in 1982, this algorithm (also known as sum-product message passing), is a message passing algorithm for computing marginal distributions on Bayesian networks (directed and acyclic graphs) and Markov random fields (undirected and cyclic graphs). Since its introduction, BP has been widely accepted as a powerful distributed algorithm in many scientific and engineering fields, including artificial intelligence, information theory, applied mathematics, signal processing and control systems [2]. Renowned applications of BP include low-density parity-check codes and turbo codes for digital communications [3]–[5], free energy approximation for statistical learning [2], satisfiability for mathematical logic [2], combinatorial optimization [6] and computer vision [7], [8]. BP also finds important applications in the area of state estimation. It is interesting to note that the famous Kalman filtering algorithm for state estimation of dynamic systems is known to be an example of BP [9]. BP has also been applied to devise distributed least-squares estimation algorithms [10], [11].

In a nutshell, BP can be roughly described as solving the following problem: Given a joint density func-

tion $P(x_1, x_2, \dots, x_n)$ for a set of discrete random variables x_1, x_2, \dots, x_n , compute the following marginals (or marginal probability density functions)

$$P_i(x_i) = \sum_{x_1 \dots x_{i-1} x_{i+1} \dots x_n} P(x_1, x_2, \dots, x_n), \quad i = 1, 2, \dots, n.$$

It is clear that brute force computation of the marginals will create combinatorics explosion because, if each x_i takes M possible values, it would require $O(M^n)$ calculations. BP circumvents this problem by exploring the structure of the function $P(\cdot)$, i.e., by analysing the network graph that interconnects the variables x_i . It is a distributed iterative algorithm. At each iteration, each variable node i (for x_i) uses all the information (messages) collected from its neighbours, updates the conditional probability density function for x_j (called message or belief), for every neighbouring node j , and transmits the resulting message to node j (called passing or propagation). Although the above description is for discrete random variables, BP also applies to continuous random variables. The so-called Gaussian BP algorithm refers to the case with Gaussian density functions [12].

For acyclic graphs (i.e., graphs without loops), it is known that the iterative process will converge in a finite number of steps, and the correct marginals will be produced [11], [13], [14]. For cyclic graphs (i.e., graphs with loops), BP is not guaranteed to converge in general, and even if it does, it does not calculate the correct marginals. Nevertheless, the wonderful and mysterious feature of BP is that for most applications, BP delivers amazingly good approximations for the marginals, despite the existence of loops [12], [15], [16]. Turbo decoding is perhaps the most successful example of such a BP application, as it delivers near-Shannon-capacity performances, despite the fact that the underlying graph is very loopy. This success has been claimed as “the most exciting and potentially important development in coding theory in many years” [17].

BP's excellent performances have inspired many researchers over the last 20 years or so to study its theoretical properties. The fundamental questions are: 1) For a cyclic network graph, under what conditions will BP iterations converge? 2) Upon convergence, how accurate are the approximate marginals? For a general cyclic graph, [18]–[22] studied the convergence condition for BP and [19], [22], [23] worked on the accuracy analysis. These references, however, only give partial answers. Moreover, the conditions offered in these references are mainly applicable to discrete random variables only. This inspires the study of Gaussian BP, as described below.

T. Sui is with Department of Control Science and Engineering, Zhejiang University, Hangzhou, 310013, China; Email: suitj@zju.edu.cn

D. E. Marelli is with School of Electrical Engineering and Computer Science, University of Newcastle, NSW 2308, Australia; Email: Damian.Marelli@newcastle.edu.au

M. Fu is with School of Electrical Engineering and Computer Science, University of Newcastle, NSW 2308, Australia. He is also a Qian-ren Scholar with Department of Control Science and Engineering and State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou, 310013, China; Email: minyue.fu@newcastle.edu.au

For the special case of Gaussian BP where the graph involves only one loop (i.e., a circle graph), satisfactory results on the two problems described above are available [24]. So far, several conditions ensuring the convergence of Gaussian BP have been proposed [14], [24]–[27]. However, there are two major drawbacks of these works:

- The condition for convergence is too difficult to check. For example, [26], [27] require solving a semi-definite program (SDP) and evaluating the spectral radius of an infinite dimensional matrix.
- They only work for nodes with scalar variables x_i .

Our study is conducted through the problem of distributed state estimation for a networked linear system with additive Gaussian noises, using the weighted least-squares criterion. The corresponding BP algorithm is known as Gaussian BP. The network has I nodes and each node i has a state vector x_i . Two types of measurements are available at each node i , the so-called *self measurement* which provides some linear measurement on x_i and the so-called *neighbour measurement* which provides some linear joint measurement between x_i and x_j with a neighbouring node j . The measurements are corrupted by additive Gaussian noises. Communications between neighbouring nodes are allowed. The goal of distributed weighted least-squares estimation is to devise a distributed iterative algorithm so that each node i will compute a good estimate of its own state x_i using its own measurements and information exchange with its neighbours. In [10], we introduced a BP-like iterative algorithm for distributed WLS estimation and showed that, for an acyclic communication graph, this algorithm converges, in a finite number of steps, to the global optimal solution (a solution obtained when the measurements for all the nodes are available). This algorithm is applied to distributed estimation of quasi-state for power systems in [11].

The purpose of this paper is to study the behaviour of Gaussian BP for cyclic graphs. The significance of this problem is of twofold. Firstly, WLS is a fundamental estimation technique with vast applications, and distributed WLS estimation is naturally needed for large-scaled networked systems when centralized solutions are not possible due to high computational loads and heavy communication burdens within the network. Secondly, solutions to this problem will be a crucial step for studying the behaviour of BP under more general settings, or for many other BP applications. Our main contribution is to show that Gaussian BP is guaranteed to converge, under a mild regularity condition. Our result significantly generalizes previous known results on BP's convergence properties, as our study allows general network graphs with cycles and network nodes with random vectors. Due to the space limit, we only provide the main findings and leave out most of the technical derivations.

II. PROBLEM FORMULATION

Suppose that there is a network graph consisting of I nodes. For each $i = 1, \dots, I$, let \mathcal{N}_i denote the set of neighbors of node i . We assume that $j \in \mathcal{N}_i$ whenever $i \in \mathcal{N}_j$. We also assume that node i can transmit its data

to every node j with $j \in \mathcal{N}_i$. This communication induces a graph \mathcal{G} , such that the (i, j) -th entry $G_{i,j}$ of its adjacency matrix G is given by

$$G_{i,j} = \begin{cases} 1, & \text{if } j \in \mathcal{N}_i, \\ 0, & \text{else.} \end{cases}$$

We assume that node i concern about its individual state $x_i \in \mathbb{R}^{n_i}$, and has a vector $h_i \in \mathbb{R}^{m_i}$ of *self measurements* and a vector $h_{i,j} \in \mathbb{R}^{m_{i,j}}$ of *neighbor measurements*, for all $j \in \mathcal{N}_i$. These measurements are given by

$$h_i = H_i x_i + \omega_i, \quad (1)$$

$$h_{i,j} = H_{i,j} x_i + H_{j,i} x_j + \omega_{i,j}, \quad (2)$$

where ω_i and $\omega_{i,j}$ are zero mean Gaussian vectors with covariances $\mathbb{E}[\omega_i \omega_i^T] = W_i > 0$ and $\mathbb{E}[\omega_{i,j} \omega_{i,j}^T] = W_{i,j} > 0$, respectively, and all these vectors are uncorrelated. Also, the measurement $h_{i,j}$ is shared by nodes i and j , i.e., $\omega_{i,j} = \omega_{j,i}$ so that $h_{i,j} = h_{j,i}$. We assume the following regularity condition:

I: For any $i = 1, 2, \dots, I$, $H_i^T W_i^{-1} H_i > 0$.

The goal of node i is to obtain an estimate of x_i using only its available measurements h_i and $h_{i,j}$, $j \in \mathcal{N}_i$, as well as communication with its neighbors in \mathcal{N}_i . A popular approach to do so is to use Gaussian BP. In this algorithm, there are two kind of nodes, namely, variable and factor nodes. To solve the above estimation problem, a variable node i is associated to each equation (1), and a factor node (i, j) to each (2). The message passed between these nodes are [27]

$$m_{i,j \rightarrow j}(x_j) = \int p(h_{i,j} | x_i, x_j) m_{i \rightarrow i,j}(x_i) dx_i, \quad (3)$$

$$m_{i \rightarrow i,j}(x_i) = p(h_i | x_i) \prod_{k \in \mathcal{N}_i \setminus j} m_{i,k \rightarrow i}(x_i). \quad (4)$$

When applied to the linear, Gaussian estimation problem defined by (1)-(2), the iterations (3)-(4) result in the procedure summarized in Algorithm 1.

Remark 1: Algorithm 1 is written in terms of the BP framework. A version of this algorithm written in terms more suitable for its implementation is given in [11].

The goal of this paper is to study the convergence property of Gaussian BP.

III. NETWORK GRAPH CONVERSION

In order to study the convergence of Gaussian BP, it is instrumental to convert the given network graph into a different one with a more convenient topology. In Section III-A we explain how a cyclic graph can be converted into an acyclic one, having an infinite number of nodes. Then, in Section III-B, we explain how to further convert the latter into a graph with a line topology.

A. Conversion of a cyclic graph into an acyclic one

We illustrate the procedure in Fig. 1, where the cyclic graph on the left is converted into the acyclic one on the right, by considering node 1 as the root node.

Let \mathcal{A} denote the infinite-sized acyclic graph resulting from the above conversion. For each node i in \mathcal{A} , let $p(i)$

Algorithm 1 Belief propagation algorithm for state estimation in system (1)-(2)

1) **Initialization:** At time $k = 0$ and variable node i , do the following two steps:

1.1) Compute

$$\begin{aligned}\alpha_i(0) &= H_i^T W_i^{-1} y_i, \\ Q_i(0) &= H_i^T W_i^{-1} H_i.\end{aligned}$$

1.2) Transmit the following information to each factor node i, j , with $j \in \mathcal{N}_i$

$$\begin{aligned}\alpha_{i \rightarrow i,j}(0) &= \alpha_i(0), \\ Q_{i \rightarrow i,j}(0) &= Q_i(0),\end{aligned}$$

2) **Main loop:** At time $N = 1, 2, \dots$ and node i , do:

2.1) Factor node i, j sends to each node $j \in \{i, j\}$

$$\begin{aligned}\alpha_{i,j \rightarrow j}(N) &= H_{i,j}^T W_{i,j \rightarrow j}^{-1}(N) y_{i,j \rightarrow j}(N), \\ Q_{i,j \rightarrow j}(N) &= H_{i,j}^T W_{i,j \rightarrow j}^{-1}(N) H_{i,j}.\end{aligned}$$

where

$$\begin{aligned}y_{i,j \rightarrow j}(N) &= y_{i,j} - H_{i,j} Q_{i \rightarrow i,j}^{-1}(N) \alpha_{i \rightarrow i,j}(N), \\ W_{i,j \rightarrow j}(N) &= W_{i,j} + H_{i,j} Q_{i \rightarrow i,j}^{-1}(N) H_{i,j}^T,\end{aligned}$$

2.2) Variable node i computes

$$\begin{aligned}\alpha_i(N) &= \alpha_i(0) + \sum_{j \in \mathcal{N}_i} \alpha_{i,j \rightarrow j}(N), \\ Q_i(N) &= Q_i(0) + \sum_{j \in \mathcal{N}_i} Q_{i,j \rightarrow j}(N),\end{aligned}$$

and

$$\begin{aligned}\hat{x}_i(N) &= Q_i^{-1}(N) \alpha_i(N), \\ \Sigma_i(N) &= Q_i^{-1}(N).\end{aligned}\quad (5)$$

2.2) Variable node i sends to each factor node i, j , with $j \in \mathcal{N}_i$

$$\begin{aligned}\alpha_{i \rightarrow i,j}(N) &= \alpha_i(0) + \sum_{k \in \mathcal{N}_i \setminus j} \alpha_{i,k \rightarrow k}(N), \\ Q_{i \rightarrow i,j}(N) &= Q_i(0) + \sum_{k \in \mathcal{N}_i \setminus j} Q_{i,k \rightarrow k}(N),\end{aligned}$$

denote the *parent* of node i (i.e., the next node found when moving towards the root) and \mathcal{S}_i denote the set of *sons* of the same node (i.e., all nodes j , with $i = p(j)$). Also, for each $N \in \mathbb{N}$, let $\mathcal{A}_N \subset \mathcal{A}$ be the sub-graph formed by nodes in \mathcal{A} which are within N hops away from the root node. The measurement equations associated to these nodes are

$$z_i = C_i \bar{x}_i + v_i, \quad (6)$$

$$z_{i,j} = C_{i,j} \bar{x}_i + C_{j,i} \bar{x}_j + v_{i,j} \quad (7)$$

for all $i \in \mathcal{A}_N$ and $j \in \mathcal{S}_i$, with $v_i \sim \mathcal{N}(0, R_i)$ and $v_{i,j} \sim \mathcal{N}(0, R_{i,j})$. The values of the quantities in (6)-(7) are given by cyclically repeating those from (1)-(2). To see

tree2.pdf

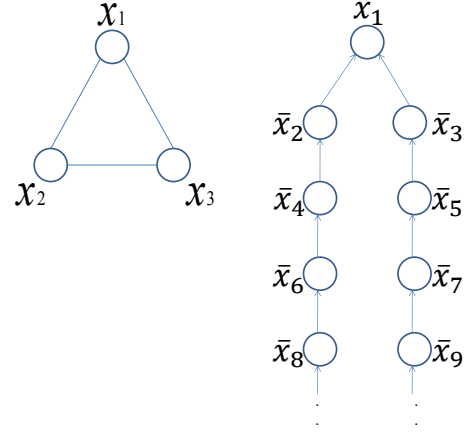


Fig. 1. Example of Conversion of a Cyclic Graph to an Acyclic One

this correspondence, we list in Table. I how the values of z_i , C_i and R_i are related to those of h_i , H_i and W_i . The correspondence of the values of $z_{i,j}$, $C_{i,j}$ and $R_{i,j}$ with $h_{i,j}$, $H_{i,j}$ and $W_{i,j}$ follows accordingly.

TABLE I

THE RELATIONSHIP BETWEEN (1)-(2) AND (6)-(7), IN THE EXAMPLE OF FIG. 1

State	Measurement	Measure Matrix	Noise Covariance
$\bar{x}_1 \in \mathbb{R}^{n_1}$	$z_1 = y_1$	$C_1 = H_1$	$R_1 = W_1$
$\bar{x}_2 \in \mathbb{R}^{n_2}$	$z_2 = y_2$	$C_2 = H_2$	$R_2 = W_2$
$\bar{x}_3 \in \mathbb{R}^{n_3}$	$z_3 = y_3$	$C_3 = H_3$	$R_3 = W_3$
$\bar{x}_4 \in \mathbb{R}^{n_3}$	$z_4 = y_3$	$C_4 = H_3$	$R_4 = W_3$
$\bar{x}_5 \in \mathbb{R}^{n_2}$	$z_5 = y_2$	$C_5 = H_2$	$R_5 = W_2$
$\bar{x}_6 \in \mathbb{R}^{n_1}$	$z_6 = y_1$	$C_6 = H_1$	$R_6 = W_1$
$\bar{x}_7 \in \mathbb{R}^{n_1}$	$z_7 = y_1$	$C_7 = H_1$	$R_7 = W_1$
$\bar{x}_8 \in \mathbb{R}^{n_2}$	$z_8 = y_2$	$C_8 = H_2$	$R_8 = W_2$
$\bar{x}_9 \in \mathbb{R}^{n_3}$	$z_9 = y_3$	$C_9 = H_3$	$R_9 = W_3$

Without loss of generality, let node 1 be the root of the graph \mathcal{A} . It follows from [11] and [22] that, at time step N , the BP algorithm gives, at node 1, the WLS estimate of \bar{x}_1 that would be obtained if \mathcal{A}_N was the whole graph. Then, the problem of studying the convergence of BP is turned into the problems of studying the convergence of the information matrix $Q_1(N)$ and the information vector $\alpha_1(N)$ of node 1, in the graph \mathcal{A}_N , as $N \rightarrow \infty$.

B. Conversion of an acyclic graph into a line

As before, let \mathcal{A}_N denote an acyclic graph having node 1 as its root and N layers. In this case, we convert \mathcal{A}_N into a line \mathcal{L}_N by collecting, for all $n = 1, \dots, N$, all the nodes in the n -th layer (\mathcal{T}_n) into a single node (node \tilde{x}_n).

Let $\mathcal{S}(t)$ denote the t -th element in the set \mathcal{S} , and $|\mathcal{S}|$ denote the number of elements in \mathcal{S} . The measurement equations associated to \mathcal{L}_N are:

$$\tilde{z}_n = \tilde{C}_n \tilde{x}_n + \tilde{v}_n, \quad (8)$$

$$\tilde{z}_{n,n+1} = \tilde{C}_{n,n+1} \tilde{x}_n + \tilde{C}_{n+1,n} \tilde{x}_{n+1} + \tilde{v}_{n,n+1}, \quad (9)$$

for each $n = 1, \dots, N$, where

$$\begin{aligned}\tilde{C}_n &= \text{diag}\{C_{\mathcal{T}_n(1)}, C_{\mathcal{T}_n(2)}, \dots, C_{\mathcal{T}_n(|\mathcal{T}_n|)}\}, \\ F_{1,i} &= \begin{bmatrix} C_{i,\mathcal{S}_i(1)}^T & C_{i,\mathcal{S}_i(2)}^T & \dots & C_{i,\mathcal{S}_i(|\mathcal{S}_i|)}^T \end{bmatrix}^T, \\ F_{2,i} &= \text{diag}\{C_{\mathcal{S}_i(1),i}, C_{\mathcal{S}_i(2),i}, \dots, C_{\mathcal{S}_i(|\mathcal{S}_i|),i}\}\end{aligned}$$

and

$$\begin{aligned}\tilde{C}_{n,n+1} &= \text{diag}\{F_{1,\mathcal{T}_n(1)}, F_{1,\mathcal{T}_n(2)}, \dots, F_{1,\mathcal{T}_n(|\mathcal{T}_n|)}\}, \\ \tilde{C}_{n+1,n} &= \text{diag}\{F_{2,\mathcal{T}_n(1)}, F_{2,\mathcal{T}_n(2)}, \dots, F_{2,\mathcal{T}_n(|\mathcal{T}_n|)}\}.\end{aligned}$$

Also, $\tilde{v}_n \sim \mathcal{N}(0, \tilde{R}_n)$ and $\tilde{v}_{n,n+1} \sim \mathcal{N}(0, \tilde{R}_{n,n+1})$, with

$$\begin{aligned}\tilde{R}_n &= \text{diag}\{R_{\mathcal{T}_n(1)}, R_{\mathcal{T}_n(2)}, \dots, R_{\mathcal{T}_n(|\mathcal{T}_n|)}\}, \\ F_{3,i} &= \text{diag}\{R_{i,\mathcal{S}_i(1)}, R_{i,\mathcal{S}_i(2)}, \dots, R_{i,\mathcal{S}_i(|\mathcal{S}_i|)}\}\end{aligned}$$

and

$$\tilde{R}_{n,n+1} = \text{diag}\{F_{3,\mathcal{T}_n(1)}, F_{3,\mathcal{T}_n(2)}, \dots, F_{3,\mathcal{T}_n(|\mathcal{T}_n|)}\}.$$

Also

$$\tilde{x}_n = \text{col}\{\bar{x}_{\mathcal{T}_n(1)}, \bar{x}_{\mathcal{T}_n(2)}, \dots, \bar{x}_{\mathcal{T}_n(|\mathcal{T}_n|)}\}.$$

Finally

$$\begin{aligned}\tilde{z}_n &= \text{col}\{z_{\mathcal{T}_n(1)}, z_{\mathcal{T}_n(2)}, \dots, z_{\mathcal{T}_n(|\mathcal{T}_n|)}\}, \\ F_{4,i} &= \begin{bmatrix} z_{i,\mathcal{S}_i(1)}^T & z_{i,\mathcal{S}_i(2)}^T & \dots & z_{i,\mathcal{S}_i(|\mathcal{S}_i|)}^T \end{bmatrix}^T\end{aligned}$$

and

$$\tilde{z}_{n,n+1} = \begin{bmatrix} F_{4,\mathcal{T}_n(1)}^T & F_{4,\mathcal{T}_n(2)}^T & \dots & F_{4,\mathcal{T}_n(|\mathcal{T}_n|)}^T \end{bmatrix}^T.$$

Notice that the change of topology from \mathcal{A}_N into \mathcal{L}_N does not change the measurement equations. Hence, the WLS estimate of node 1 in \mathcal{L}_N is equivalent to that in \mathcal{A}_N .

IV. CONVERGENCE CONDITION FOR BP

From (5), the BP algorithm computes the estimate $\hat{x}_i(k)$ by multiplying the inverse of the information matrix $Q_i(k)$ with the information vector $\alpha_i(k)$. Hence, we need to study the convergence of two quantities. In Section IV-A we study the convergence of the information matrix $Q_i(k)$, and we do the same for the state estimate $\hat{x}_i(k)$ in Section IV-B. Without loss of generality, we focus our analysis on the convergence at node 1.

Notation 1: For any graph \mathcal{C} (either cyclic or acyclic) we use $\hat{x}_i(\mathcal{C})$, $Q_i(\mathcal{C})$ and $\alpha_i(\mathcal{C})$ to denote the estimate, information vector and information matrix resulting from running the WLS algorithm on the whole graph \mathcal{C} .

Definition 1: For $P, Q > 0$, we define the Riemannian Distance between P and Q by

$$\delta(P, Q) = \sqrt{\sum_i \log^2 \text{eig}_i(PQ^{-1})}.$$

Our convergence results depend on the system's parameters in (1)-(2) in a rather complicated manner. Nevertheless, they are fully determined by the knowledge of the following set of constants.

Notation 2: For a matrix A , we use $\sigma_{\max}(A)$, $\sigma_{\min}(A)$ to denote its maximum and minimum singular value, respectively. We also define

$$\bar{n} = \max_i \dim x_i, \quad \bar{m} = \max_{i,j} \{\dim h_i, \dim h_{i,j}\},$$

$$\bar{r} = \max_{i,j} \{\|W_i\|, \|W_{i,j}\|\},$$

$$\underline{r} = \min_{i,j} \{\sigma_{\min}(W_i), \sigma_{\min}(W_{i,j})\},$$

$$\bar{h} = \max_{i,j} \{\|H_i\|, \|H_{i,j}\|\},$$

$$\underline{h} = \min_{i,j} \{\sigma_{\min}(H_i), \sigma_{\min}(H_{i,j})\},$$

$$\bar{z} = \max_{i,j} \{\|h_i\|_{\infty}, \|h_{i,j}\|_{\infty}\},$$

$$\bar{u} = \max_i |\mathcal{N}_i| - 1, \quad \bar{\xi} = \max_i \log \sigma_{\max}[I + M_i],$$

$$M_i = \left(\sum_{j \in \mathcal{N}_i} H_{i,j}^T W_{i,j}^{-1} H_{i,j} \right) (H_i^T W_i^{-1} H_i)^{-1}.$$

A. Convergence Condition for information matrix

We give an intuitive explanation of our approach. In view of the graph conversion described in Section III-A, studying the limit value of $Q_1(N)$ is equivalent to study the limit of $Q_1(\mathcal{A}_N)$, as N tends to infinity. Also, in view of the conversion in Section III-B, we in turn have that $Q_1(\mathcal{A}_N) = Q_1(\mathcal{L}_N)$. Then, as the first step towards our main result, we have the following lemma, which applies to a graph \mathcal{L}_N with line topology.

Lemma 1: For each N and t

$$\|Q_1^{-1}(\mathcal{L}_{N+1}) - Q_1^{-1}(\mathcal{L}_N)\| \leq (e^{\rho^{N-1}\delta_N} - 1) \beta_1^{-1},$$

where,

$$\delta_N = \delta \left(\tilde{C}_N^T \tilde{R}_N^{-1} \tilde{C}_N, \right.$$

$$\left. \tilde{C}_N^T \tilde{R}_N^{-1} \tilde{C}_N + \tilde{C}_{N,N+1}^T \tilde{R}_{N,N+1}^{-1} \tilde{C}_{N,N+1} \right),$$

$$\rho = \frac{\alpha_1}{\alpha_1 + \beta_1} \frac{\alpha_2}{\alpha_2 + \beta_2},$$

with

$$\alpha_1 = \frac{(\bar{u} + 1)\bar{h}^2}{\underline{r}}, \quad \beta_1 = \frac{\bar{h}^2}{\bar{r}}, \quad \alpha_2 = \frac{\bar{h}^2}{\beta_1}, \quad \beta_2 = \underline{r}.$$

Lemma 1 states that the contribution to the information matrix at node 1, from nodes which are far away, decays exponentially with the number of hops. The constants ρ and β_1 are independent of N and the root node (i.e., whose information matrix we are studying). On the other hand, δ_N depends on the definition of \mathcal{L}_N , which in turn depends on the root node. Our main result, stated below, deals with this situation.

Theorem 1: For every $N \in \mathbb{N}$,

$$\|Q_1^{-1}(N-1) - Q_1^{-1}(N)\| \leq \beta_1^{-1} (e^{\bar{\rho}^{N-1}\bar{\delta}} - 1),$$

where $\bar{\rho} = \rho\sqrt{\bar{u}}$ and $\bar{\delta} = \sqrt{\bar{n}(\bar{u} + 1)}\bar{\xi}$. Moreover, if $\bar{\rho} < 1$ and $N > 1 + \lceil \log_{\bar{\rho}} \frac{1}{\bar{\delta}} \rceil$, then

$$\|Q_1^{-1}(N-1) - Q_1^{-1}(N)\| \leq \frac{2\bar{\delta}}{\beta_1} \bar{\rho}^{N-1}.$$

Theorem 1 states that the increments of $Q_1^{-1}(N)$ vanish exponentially. Then, the desired convergence condition follows as a corollary of the this result.

Corollary 1: If $\bar{\rho} < 1$, there exists $\bar{Q}_1 > 0$ such that

$$\lim_{N \rightarrow \infty} Q_1(N) = \bar{Q}_1$$

and the convergence is exponential.

B. Convergence Condition for the estimate

In this section we study the convergence of $\hat{x}_1(k)$. To this end, we separate our study in two parts. In Section IV-B.1, we study the convergence in a graph with line topology, and in Section IV-B.2 we extend this result to a graph with arbitrary topology.

1) *Convergence for a graph with line topology:* Consider a graph \mathcal{L}_N with line topology, whose measurement equations are given by (8)-(9). Let $y_i^T = [\tilde{z}_i^T, \tilde{z}_{i,i+1}^T]$, $w_i^T = [\tilde{v}_i^T, \tilde{v}_{i,i+1}^T]$ and

$$A_{ii} = \begin{bmatrix} \tilde{C}_i \\ \tilde{C}_{i,i+1} \end{bmatrix}, \quad A_{i,i+1} = \begin{bmatrix} 0 \\ \tilde{C}_{i+1,i} \end{bmatrix}, \quad (10)$$

$$S_i = \begin{bmatrix} \tilde{R}_i & 0 \\ 0 & \tilde{R}_{i,i+1} \end{bmatrix}. \quad (11)$$

Then, (8)-(9) becomes

$$y_i = A_{ii}\tilde{x}_i + A_{i,i+1}\tilde{x}_{i+1} + w_i,$$

with $w_i \sim \mathcal{N}(0, S_i)$. Let also $\mathbf{x}_N^T = [\tilde{x}_1^T, \dots, \tilde{x}_N^T]$, $\mathbf{y}_N^T = [y_1^T, \dots, y_N^T]$, $\mathbf{w}_N^T = [w_1^T, \dots, w_N^T]$ and

$$\mathbf{A}_N = \begin{bmatrix} A_{11} & A_{12} & 0 & \cdots & 0 \\ 0 & A_{22} & A_{23} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & A_{N-1,N-1} & A_{N-1,N} \\ 0 & \cdots & \cdots & 0 & A_{N,N} \end{bmatrix}.$$

We then have

$$\mathbf{y}_N = \mathbf{A}_N \mathbf{x}_N + \mathbf{w}_N,$$

with $\mathbf{w}_N \sim \mathcal{N}(0, \mathbf{S}_N)$ and $\mathbf{S}_N = \text{diag}\{S_1, \dots, S_N\}$.

The WLS estimator $\hat{\mathbf{x}}_N$ of \mathbf{x}_N is given by

$$\hat{\mathbf{x}}_N = \mathbf{Q}_N^{-1} \mathbf{q}_N,$$

where $\mathbf{q}_N^T = \mathbf{A}_N^T \mathbf{S}_N^{-1} \mathbf{y}_N = [q_1, \dots, q_N]$ with

$$q_i = \begin{cases} A_{ii}^T S_i^{-1} y_i, & i = 1, \\ A_{ii}^T S_i^{-1} y_i + A_{i-1,i}^T S_{i-1}^{-1} y_{i-1}, & i > 1, \end{cases} \quad (12)$$

and the (i, j) -th entry $Q_{i,j}$ of $\mathbf{Q}_N = \mathbf{A}_N^T \mathbf{S}_N^{-1} \mathbf{A}_N$ given by

$$Q_{i,i} = \begin{cases} A_{ii}^T S_i^{-1} A_{ii}, & i = 1, \\ A_{ii}^T S_i^{-1} A_{ii} + A_{i-1,i}^T S_{i-1}^{-1} A_{i-1,i}, & i > 1, \end{cases}$$

$$Q_{i,i+1} = A_{ii}^T S_i^{-1} A_{i,i+1},$$

$$Q_{i,j} = 0, \quad |i - j| \geq 2.$$

Let

$$\Sigma_N = \mathbf{Q}_N^{-1}.$$

Denote the $i - j$ block of Σ_N by $[\Sigma_N]_{i,j}$. From the inverse formula for band matrices [28], it follows that the first row of Σ_N is given by

$$[\Sigma_N]_{1,j} = \left(\prod_{k=1}^{j-1} \Delta_k^{-1} Q_{k,k+1} \right) \Phi_j^{-1}(N),$$

$$\Phi_j(N) = \Gamma_j(N) - Q_{j-1,j} \Delta_{j-1}^{-1} Q_{j,j-1}, \quad (13)$$

with

$$\Delta_k = \begin{cases} Q_{kk}, & k = 1, \\ Q_{kk} - Q_{k,k-1} \Delta_{k-1}^{-1} Q_{k-1,k}, & k > 1, \end{cases} \quad (14)$$

$$\Gamma_k(N) = \begin{cases} Q_{kk}, & k = N, \\ Q_{kk} - Q_{k,k+1} \Gamma_{k+1}^{-1}(N) Q_{k+1,k}, & k < N, \end{cases} \quad (15)$$

for any $j = 1, 2, \dots, N$. Then, the first entry $[\hat{\mathbf{x}}_N]_1$ of $\hat{\mathbf{x}}_N$ is given by

$$[\hat{\mathbf{x}}_N]_1 = \sum_{j=1}^N [\Sigma_N]_{1,j} q_j. \quad (16)$$

By running the BP algorithm on the graph \mathcal{L}_N , it will produce, at node 1 and step N , the WLS estimate [11], i.e.,

$$\hat{x}_1(N) = [\hat{\mathbf{x}}_N]_1.$$

It then follows from (16) that

$$\begin{aligned} & \left\| \hat{x}_1(N) - \hat{x}_1(N-1) \right\| \\ & \leq \sum_{j=1}^{N-1} \left\| [\Sigma_N]_{1,j} - [\Sigma_{N-1}]_{1,j} \right\| \|q_j\| + \left\| [\Sigma_N]_{1,N} \right\| \|q_N\|. \end{aligned} \quad (17)$$

The main result of this section is given in Lemma 2. It states the upper bound of the difference in (17), as N tends to infinity.

Lemma 2: For any $N > 2\lceil -\log_{\lambda_1} \delta_N \rceil$,

$$\left\| \hat{x}_1(N) - \hat{x}_1(N-1) \right\| \leq \eta_N \left(c_{1,N} \lambda_1^{N/2} + c_{2,N} \lambda_2^{N/\alpha} \right),$$

where

$$\lambda_1 = \frac{\bar{\pi}_1 \bar{\pi}_2}{(\bar{\pi}_1 + \underline{\pi}_1)(\bar{\pi}_2 + \underline{\pi}_2)}, \quad \lambda_2 = \frac{\sqrt{\bar{q}} - \sqrt{q}}{\sqrt{\bar{q}} + \sqrt{q}},$$

$$c_{1,N} = \frac{2d\delta_N}{\lambda_1 \bar{q}(d-1)}, \quad c_{2,N} = \frac{\bar{q} - q}{\bar{q} \lambda_2 (1 - \lambda_2)},$$

$$\eta_N = \max_{n \leq N} \|q_n\|, \quad \alpha = \max \left\{ -\frac{1}{2} \log_d \lambda_1, 2 \right\}.$$

and

$$d = \frac{\bar{q}}{q \lambda_1}, \quad \bar{q} = (2\bar{u} + 3) \bar{h}^2 \bar{r}^{-1}, \quad q = \underline{h}^2 \bar{r}^{-1},$$

$$\bar{\pi}_1 = \underline{r}^{-1}(\bar{u} + 1) \bar{h}^2, \quad \bar{\pi}_2 = \frac{\bar{h}^2 \bar{r}}{\underline{h}^2}, \quad \underline{\pi}_1 = \bar{r}^{-1} \underline{h}^2, \quad \underline{\pi}_2 = \underline{r},$$

2) *Convergence for a graph with arbitrary topology:* We give an overview of our approach. For each N , the value of $\hat{x}_1(N)$ obtained by running BP algorithm on the original cyclic graph \mathcal{G} , equals the estimate $\hat{\hat{x}}_1(N)$ that would be obtained if the BP algorithm is applied to the acyclic graph \mathcal{A}_N , i.e., $\hat{x}_1(N) = \hat{\hat{x}}_1(N)$. This in turn equals the estimate $\hat{\hat{x}}_1(N)$ that would be obtained in the graph \mathcal{L}_N , if the latter is built as explained in Section III-B. Hence, $\hat{x}_1(N) = \hat{\hat{x}}_1(N)$.

Since \mathcal{L}_N has a line topology, we can apply Lemma 2 here. This requires finding uniform bounds (in the sense of being independent of N) for the two quantities in the statement of that lemma, which depend on N , namely, $c_{1,N}$ and η_N . These bounds are obtained as follows

$$c_{1,N} \leq \bar{\psi} \bar{u}^{N/2}, \quad \eta_N \leq \bar{\eta} \bar{u}^{N/2},$$

with

$$\bar{\psi} = \frac{2d}{\lambda_1 \bar{q}(d-1)} \bar{\xi} \sqrt{\bar{n}(\bar{u}+1)},$$

$$\bar{\eta} = \frac{\bar{\varepsilon} z \sqrt{8\bar{m}(\bar{u}+1)}}{r}.$$

Our main result follows then straightforwardly from Lemma 2 and the above bounds.

Theorem 2: If $\bar{u}\lambda_1 < 1$, then, for any $N > 2 \left\lceil \log_{\bar{u}\lambda_1} (\lambda_1 / \bar{\xi} \sqrt{\bar{n}(\bar{u}+1)}) \right\rceil$, we have

$$\|\hat{x}_1(N) - \hat{x}_1(N-1)\| \leq \bar{\chi} \kappa^N,$$

with $\bar{\chi} = \bar{\eta} (\bar{\psi} + c_2)$ and $\kappa = \max \left\{ \bar{u} \sqrt{\lambda_1}, \sqrt{\bar{u}} \lambda_2^{1/\alpha} \right\}$.

Based on the exponential decay stated in Theorem 2, we can derive the desired convergence condition.

Corollary 2: If $\kappa < 1$, then there exists $\hat{x}_{1,\infty}$ such that

$$\lim_{N \rightarrow \infty} \hat{x}_1(N) = \hat{x}_{1,\infty},$$

and the convergence is exponential.

Remark 2: The condition $\bar{u}\lambda_1 < 1$ in Theorem 2 is always satisfied if $\kappa < 1$. Thus, it does not affect the decay condition in Corollary 2.

V. CONCLUSION

We have reported in this paper the convergence property of the well-known Gaussian BP algorithm. This is carried out via the problem of distributed sate estimation for a networked linear system. This is an important step in the complete understanding of the behaviour of Gaussian BP. We hope that our result will generate more interest in this powerful statistical learning tool and will encourage more application of the algorithm in distributed estimation and control.

REFERENCES

- [1] J. Pearl, "Reverend bayes on inference engines: A distributed hierarchical approach," *Proceedings of the Second National Conference on Artificial Intelligence*, pp. 133–136, 1982.
- [2] A. Braunstein, R. Mézard, and R. Zecchina, "Survey propagation: An algorithm for satisfiability," *Random Structures & Algorithms*, vol. 27, no. 2, pp. 201–226, 2005.
- [3] R. J. McEliece, D. J. C. MacKay, and J.-F. Cheng, "Turbo decoding as an instance of pearl's belief propagation algorithm," *Selected Areas in Communications, IEEE Journal on*, vol. 16, no. 2, pp. 140–152, 1998.

- [4] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," *Communications, IEEE Transactions on*, vol. 44, no. 10, pp. 1261–1271, 1996.
- [5] M. Fu, "Stochastic analysis of turbo decoding," *IEEE Transactions on Information Theory*, vol. 54, no. 1, pp. 81–100, 2005.
- [6] A. Braunstein and R. Zecchina, "Survey propagation as local equilibrium equations," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2004, no. 06, p. P06007, 2004.
- [7] J. Sun, N.-N. Zheng, and H.-Y. Shum, "Stereo matching using belief propagation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, no. 7, pp. 787–800, 2003.
- [8] K. Tanaka, "Statistical-mechanical approach to image processing," *Journal of Physics A: Mathematical and General*, vol. 35, no. 37, p. R81, 2002.
- [9] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, 2001.
- [10] D. E. Marelli and M. Fu, "Distributed weighted least-squares estimation with fast convergence for large-scale systems," *Automatica*, vol. 51, pp. 27–39, 2015.
- [11] X. Tai, Z. Lin, M. Fu, and Y. Sun, "A new distributed state estimation technique for power networks," in *American Control Conference*, 2013, pp. 3338–3343.
- [12] B. J. Frey, "Graphical models for pattern classification, data compression and channel coding," 1998.
- [13] J. Pearl, *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufman, 1988.
- [14] Y. Weiss and W. T. Freeman, "Correctness of belief propagation in gaussian graphical models of arbitrary topology," *Neural computation*, vol. 13, no. 10, pp. 2173–2200, 2001.
- [15] K. P. Murphy, Y. Weiss, and M. I. Jordan, "Loopy belief propagation for approximate inference: An empirical study," in *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1999, pp. 467–475.
- [16] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael, "Learning low-level vision," *International journal of computer vision*, vol. 40, no. 1, pp. 25–47, 2000.
- [17] R. J. McEliece, E. R. Rodemich, and J.-F. Cheng, "The turbo decision algorithm," in *Proceedings of The Annual Allerton Conference on Communication Control and Computing*, vol. 33. UNIVERSITY OF ILLINOIS, 1995, pp. 366–379.
- [18] S. C. Tatikonda and M. I. Jordan, "Loopy belief propagation and gibbs measures," in *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 2002, pp. 493–500.
- [19] S. C. Tatikonda, "Convergence of the sum-product algorithm," in *Information Theory Workshop, 2003. Proceedings. 2003 IEEE*. IEEE, 2003, pp. 222–225.
- [20] A. T. Ihler, J. W. Fisher, and A. S. Willsky, "Message errors in belief propagation," in *Advances in Neural Information Processing Systems*, 2004, pp. 609–616.
- [21] J. M. Mooij and H. J. Kappen, "Sufficient conditions for convergence of the sum-product algorithm," *Information Theory, IEEE Transactions on*, vol. 53, no. 12, pp. 4422–4437, 2007.
- [22] A. T. Ihler, J. Iii, and A. S. Willsky, "Loopy belief propagation: Convergence and effects of message errors," in *Journal of Machine Learning Research*, 2005, pp. 905–936.
- [23] T. Heskes, "On the uniqueness of loopy belief propagation fixed points," *Neural Computation*, vol. 16, no. 11, pp. 2379–2413, 2004.
- [24] D. M. Malioutov, J. K. Johnson, and A. S. Willsky, "Walk-sums and belief propagation in Gaussian graphical models," *The Journal of Machine Learning Research*, vol. 7, pp. 2031–2064, 2006.
- [25] C. C. Moallemi and B. Van Roy, "Convergence of min-sum message-passing for convex optimization," *IEEE Transactions on Information Theory*, vol. 56, no. 4, pp. 2041–2050, 2010.
- [26] Q. Su and Y.-C. Wu, "Convergence analysis of the variance in Gaussian belief propagation," *IEEE Transactions on Signal Processing*, vol. 62, pp. 5119–5131, 2014.
- [27] —, "On convergence conditions of Gaussian belief propagation," *IEEE Transactions on Signal Processing*, vol. 63, pp. 1144–1155, 2015.
- [28] G. Meurant, "A review on the inverse of symmetric tridiagonal and block tridiagonal matrices," *SIAM Journal on Matrix Analysis and Applications*, vol. 13, no. 3, pp. 707–728, 1992.