

Structured Variational Methods for Distributed Inference: Convergence Analysis and Performance-Complexity Tradeoff

Yanbing Zhang and Huaiyu Dai
Department of Electrical and Computer Engineering
NC State University
Raleigh, N.C. USA
{yzhang, Huaiyu_Dai}@ncsu.edu

Abstract –In this paper, the asymptotic performance of a recently proposed distributed inference framework, structured variational methods, is investigated. We first distinguish the intra- and inter-cluster inference algorithms as vertex and edge processes respectively. Their difference is illustrated, and convergence rate is derived for the intra-cluster inference procedure which is based on an edge process. Then, viewed as a mixed vertex-edge process, the overall performance of structured variational methods is characterized via the coupling approach. Tradeoff between complexity and performance of this algorithm is also addressed, which provides insights for network design and analysis.

I. INTRODUCTION

Large-scale networked systems of intelligent devices are playing an increasingly important role in our life. In such systems, finding solutions in a distributed fashion, in the absence of a central coordinator, is of great importance. Various distributed inference algorithms, such as belief propagation (BP) [1], consensus propagation (CP, a special case of Gaussian BP) [2] and Gossip algorithm [3], have been proven efficient, robust and scalable.

Variational method (mean field (MF) approach) is also attractive for distributed inference due to its computational simplicity, nevertheless suffers from slow convergence (equivalently inferior inference performance in given time). In [4], a general framework of structured variational methods was proposed to simultaneously exploit the simplicity of variational methods (for inter-cluster processing) and the accuracy of BP algorithms (for intra-cluster processing). Its advantages in wireless networks were testified by simulations. One key observation there is: a tradeoff between communication complexity (energy consumption) and inference performance can be achieved by appropriately selecting the cluster size and associated structure. This motivates us to further quantify this tradeoff in hybrid wireless networks.

In linear distributed algorithms, typically a stochastic weight matrix is employed, which is conformant to certain underlying graphical structures (network topology). Hence the convergence of such algorithms is closely related to the mixing time of a random walk on the corresponding graph. According to [2], random walks on graphs can be categorized as *vertex process*-based or *edge process*-based ones. The essential difference between these two is that the former is a process on nodes that transitions along edges and is allowed to “backtrack”, while the later is a process on *directed* edges that transitions towards nodes where “backtrack” is forbidden. As we will see, distributed algorithm derived from the variational method can be characterized by a vertex process, typically involving reversible Markov chains; while belief propagation and its variants correspond to edge processes, typically involving non-reversible Markov chains.

Even though quite a few techniques exist for analyzing the convergence of reversible Markov chains, including spectral theory, conductance, canonical paths and multi-commodity flow (see [8] and the references therein), few of them can be successfully applied to non-reversible cases. [5] analyzes a non-reversible random walk in the one-dimensional chain through a direct probabilistic approach. A study on the convergence properties of consensus propagation is given in [2] through function mapping and matrix analysis; an explicit result on convergence time is derived for the cycle, with conjectures given for higher-dimensional tori. Structured variational methods, as we will formulate in Section II, actually correspond to mixed vertex-edge processes involving hybrid Markov chains, entailing even more difficulties on analysis. In this paper, we use a “divide and conquer” strategy to investigate its performance: first we analyze the convergence of the intra-cluster edge process, where we derive an upper bound on the mixing time for the two-dimensional (2-d) torus; then we explore the coupling technique [9] to combine the results for edge and vertex processes to obtain a characterization on the overall performance. As a result, the performance-complexity tradeoff in structured variational methods is analytically addressed.

The rest of this paper is organized as follows. Section II gives a brief introduction on structured variational methods and formulates the problem. Convergence analysis for the intra-cluster process is given in Section III. Section IV explores the overall convergence rate, together with the tradeoff between complexity and performance. Finally, concluding remarks are given in Section V.

II. STRUCTURED VARIATIONAL METHODS AND PROBLEM FORMULATION

A. Structured Variational Methods

For concreteness of discussion, consider a Gaussian pairwise¹ Markov random field (MRF) X on an undirected graph $G = (V, E)$, with V and E denoting the vertex and edge set respectively. Each node $i \in V$ is associated with a spatial component X_i of X , assumed a priori to have zero mean and covariance matrix

$$\Sigma_X = \sigma_s^2 \begin{bmatrix} 1 & \rho_{12} & \cdots & \rho_{1|V|} \\ \rho_{21} & 1 & \cdots & \rho_{2|V|} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{|V|1} & \rho_{|V|2} & \cdots & 1 \end{bmatrix}, \quad (1)$$

where σ_s^2 is the prior variance of each component, and ρ_{ij} denotes the correlation coefficient between node i and j . Every node i makes a noisy linear observation

$$y_i = H_i x_i + \xi_i, \quad i = 1, \dots, |V|, \quad (2)$$

where channel gain H_i is assumed known, and noise ξ_i is Gaussian with zero mean and variance σ^2 . Given the observation vec-

This work was supported in part by the US National Science Foundation under Grant CCF-0515164, CNS-0721815 and CCF-0830462.

¹ MRF with higher order cliques can always be converted into an equivalent pairwise MRF.

tor $y = [y_1, \dots, y_{|V|}]^T$, the posterior probability $P(X | y)$ is Gaussian distributed as

$$P(X | y) \sim \mathcal{N}(F^{-1}Hy / \sigma^2, F^{-1}), \quad (3)$$

where $H = \text{diag}(H_i)$, $F = [F_{ij}]_{|V| \times |V|} = \Sigma_x^{-1} + H^2 / \sigma^2$.

With the variational method, the posterior probability (3) is approximated by a simpler ‘‘variational’’ distribution

$$Q(X) = \prod_{i=1}^N Q_i(X_i) = \alpha \prod_{i=1}^N \exp\left\{-\frac{(X_i - \mu_i)^2}{\sigma_i^2}\right\}, \quad (4)$$

where μ_i and σ_i^2 are the posterior mean and variance respectively. Variational methods yield an iterative form to estimate μ_i [4]:

$$\mu_{\text{MF},i}^{(n)} = \left[y_i / H_i \sigma^2 - \sum_{j \in \Gamma(i)} F_{ij} \mu_{\text{MF},j}^{(n-1)} \right] / F_{ii}, \quad (5)$$

where $\Gamma(i)$ stands for the set of neighboring nodes of i .

On the other hand, if we consider the correlation in the network, belief propagation can be pursued for more accurate inference. All messages are Gaussian distributed: iterative update for the mean μ_{ij} of the message from node i to node j is given by

$$\mu_{ij}^{(n)} = \frac{\Sigma_{ij}^{(n)} \rho_{ij} \left(H_i y_i / \sigma^2 + \sum_{k \in \Gamma(i) \setminus \{j\}} (\Sigma_{ki}^{(n-1)})^{-1} \mu_{ki}^{(n-1)} \right)}{1 + \sigma_s^2 (1 - \rho_{ij}^2) \left(H_i^2 / \sigma^2 + \sum_{k \in \Gamma(i) \setminus \{j\}} (\Sigma_{ki}^{(n-1)})^{-1} \right)}, \quad (6)$$

where Σ_{ij} is the message variance, whose update is omitted here in the interest of space (see [4] for details). After all the messages from neighbors are received, the posterior mean of node i is computed by

$$\mu_{\text{BP},i}^{(n)} = \sum_{j \in \Gamma(i)} (\Sigma_{ji}^{(n)})^{-1} \mu_{ji}^{(n)} / \sum_{j \in \Gamma(i)} (\Sigma_{ji}^{(n)})^{-1}. \quad (7)$$

Although attractive for its computational simplicity, the naive mean field approach (5) may not yield sufficient accuracy, or equivalently, fast convergence. BP algorithm converges faster at a cost of higher computation and communication complexity, or more energy consumption in practical use. A natural idea for improvement is to integrate these two approaches, thus the simplicity of the variational method and the fast convergence of the BP algorithm can be exploited simultaneously. In particular, if some probabilistically tractable substructures (clusters) C_m , $m = 1, \dots, M$, can be identified, these substructures can be handled by BP, while the mean field approach can be employed for information propagation between the substructures. This approach is referred to as the structured variational method, or structured mean field (SMF) approach. In SMF, intra-cluster inference continues to update as (6); inter-cluster updating is conducted on the ‘‘gateway’’ nodes between clusters. For a gateway node i in cluster C_i , following the MF approach, the update is given as [4]

$$y_i^{(n)} = y_i^{(n-1)} - \sigma^2 \sum_{j \in \text{MB}(C_i)} \rho_{ij} \mu_{\text{BP},i}^{(n-1)}, \quad i \in C_i, \quad (8)$$

where $\text{MB}(C_i)$ is the Markov blanket of cluster C_i . That is, gateway nodes use the estimates of their neighbors in the Markov blanket to ‘‘update’’ observations, and use these ‘‘new’’ observations for the next round intra-cluster inference.

B. Vertex, Edge and Mixed Process

It has been noted that the message variance iteration converges much faster than the message mean iteration in the BP algorithm. Therefore the convergence behavior of the posterior mean (7) is mainly determined by the message mean iteration (6), assuming that variance has already converged to some $\{\Sigma_{ij}\}$ [2][4].

Both updates in (5) and (6) (with the above assumption) can be written in a vector-matrix form as

$$\boldsymbol{\mu}^{(n)} = A\boldsymbol{y} + B\hat{P}\boldsymbol{\mu}^{(n-1)}, \quad (9)$$

where A, B are relevant coefficient matrices, while \hat{P} defines a stochastic, irreducible and aperiodic Markov chain. The \hat{P} for (5) is a $|V| \times |V|$ matrix defined on the vertex set with the entries

$$P_{ij} = \begin{cases} F_{ij} / \sum_{j \in \Gamma(i)} F_{ij}, & j \in \Gamma(i) \\ 0, & \text{otherwise;} \end{cases} \quad (10)$$

while for (6) it turns out to be a $2|E| \times 2|E|$ matrix defined on the set of directed edges whose entries are

$$P_{e',e} = \begin{cases} \frac{K_{s(e)d(e) \setminus \{d(e)\}}}{\sum_{e':s(e') \in \Gamma(s(e)) \setminus \{d(e)\}} K_{s(e)d(e) \setminus \{d(e)\}}}, & s(e) = d(e') \\ & \text{but } s(e') \neq d(e) \\ 0, & \text{otherwise,} \end{cases} \quad (11)$$

where $s(e)$ and $d(e)$ denote the source and destination node of edge e , and $K_{ki \setminus \{j\}}$ admits the form

$$K_{ki \setminus \{j\}} = \sigma^2 \sigma_s^2 (1 - \rho_{ij}^2) (\Sigma_{ki})^{-1} / \left[\sigma^2 + \sigma_s^2 (1 - \rho_{ij}^2) \right].$$

These two schemes correspond to vertex process and edge process respectively.

Fig. 1 illustrates the distinction between a vertex process and an edge process. As (a) shows, the states in a vertex process are nodes (the circles), while the allowable (two-way) transition between the states is determined by the undirected edges. Contrastively, the states in an edge process are represented by the directed edges (the arrows in (b)), and the transitions are guided by the directions that the arrows point to. In other words, the transition can only occur between the states whose pointed directions are not against each other, which corresponds to the condition $s(e) = d(e')$ but $s(e') \neq d(e)$ in (11), and the rule in BP that the message from one neighbor only contributes to the new messages sent to other neighbors (but not back to itself) (c.f. (6)). For structured variational methods, we constrain the edge process only within clusters, and employ the vertex process to exchange information between clusters. This leads to a mixed vertex-edge process model as shown in (c).

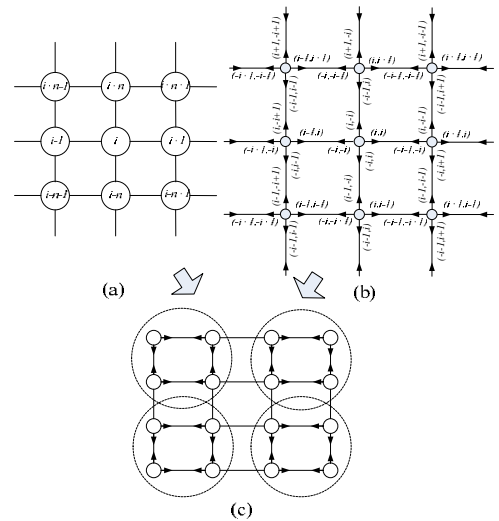


Fig. 1 Vertex Process (a), Edge Process (b) and Mixed Process (c)

III. CONVERGENCE RATE OF EDGE PROCESS

For a Markov chain \hat{P} on the state space Ω with stationary distribution π , the mixing time

$$t_{\text{mix}}(\varepsilon) = \max_i \inf \left\{ t : \|\hat{P}^t(i, \cdot) - \pi\|_1 / 2 \leq \varepsilon \right\} \quad (12)$$

specifies the time that \hat{P} takes to converge to the ε -vicinity of its stationary distribution.

The convergence behavior of vertex processes has been well studied in the context of reversible Markov chains. In particular, it is not difficult to prove that the mixing time of a reversible Markov chain on a 2-d $n \times n$ torus is $t_{\text{mix}} = O(n^2)$ [12], which characterizes the convergence time of vertex processes and thus the variational method. However, the performance of edge processes, which in general involves non-reversible Markov chains, is still largely unexplored. And to the best of our knowledge, there is no formal convergence discussion on the mixed model.

It has been observed that certain non-reversible chains mix substantially faster than corresponding reversible chains, by suppressing the diffusive behavior of the latter [5][6]. Motivated by this finding, we have explored the idea of constructing non-reversible chains to achieve fast distributed averaging in wireless networks, and proposed a class of Location-Aided Distributed Averaging (LADA) algorithms [7]. But all of these works only focus on some specific values of the probability that the random walks make turns (typically on the order of 1/scale of the graph).

For many other distributed algorithms, including BP and CP, however, the turning probabilities of associated chains are not tunable, but determined by the algorithms and applications. A conjecture is put forth in [2] that the convergence time of consensus propagation (with corresponding turning probability a constant) on a 2-d $n \times n$ torus is $O(n^{3/2})$. In this section, we verify this conjecture through deriving a general upper bound for the mixing time of edge processes on a 2-d torus, assuming a turning probability $q(n)/n$, where $q(n)$ satisfies $\lim_{n \rightarrow \infty} q(n) \rightarrow \infty$ and $q(n)/n \leq 1/3$ (for the case that $q(n)$ is constant, refer to [5][7]). This result will also facilitate the analysis of the overall performance of structured variational methods in the next section.

We begin with citing a result from [7], which applies to general Markov chains.

Lemma 1: For any irreducible and aperiodic Markov chain \hat{P} with stationary distribution π ,

$$t_{\text{mix}}(\varepsilon) \leq \left[\log(\varepsilon^{-1}) / \log((1-c)^{-1}) + 1 \right] t_{\text{fill}}(c), \quad (13)$$

where $t_{\text{fill}}(c) = \max_i \inf \{ t : \hat{P}^t(i, \cdot) \geq c\pi \}$, $0 < c < 1$.

As shown in Fig. 2, a two-tuple $(s_0, s_1) \in \{-n, \dots, -1, 1, \dots, n\}^2$ is used to represent an edge process on an $n \times n$ torus. Specifically, given node $(1,1)$ on the left-bottom corner, and (n,n) on the right-top corner, four outgoing edges of node (i, j) pointing to the East, North, West and South directions are respectively labeled as (i, j) , $(j, -i)$, $(-i, -j)$ and $(-j, i)$. The states are only allowed to turn left or turn right with the probability $q(n)/n$ and move forward with the probability $1 - 2q(n)/n$, but can't backtrack (e.g., the transition from (i, j) to $(-i-1, -j)$ is forbidden while the other three allowed).

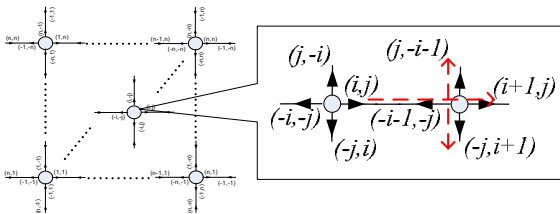


Fig. 2 State Labeling of Edge Process on a 2-d Torus

Due to our state representation, it can be verified that the state evolution (whether horizontal or vertical) admits:

$$\text{Moving Forward} \Rightarrow (s_0^{i+1} = s_0^i + 1, s_1^{i+1} = s_1^i), \quad (14)$$

$$\text{Turning Left} \Rightarrow (s_0^{i+1} = s_1^i, s_1^{i+1} = -s_0^i - 1), \quad (15)$$

$$\text{Turning Right} \Rightarrow (s_0^{i+1} = -s_1^i, s_1^{i+1} = s_0^i + 1). \quad (16)$$

Without loss of generality, assume the random walk starts from state $(s_0^0, s_1^0) = (a, b)$. Let $T_0 = 0, T_1, T_2, \dots$ be the time instances that the random walk makes turns, and $D_i = L/R$ be the corresponding turning direction (Left or Right) at time i , then for the time $t \in [T_k, T_{k+1})$, with k being the total number of turnings before time t , the destination state evolves as

$$(s_0^t, s_1^t) = \begin{cases} (t + s_1^{T_{k-1}} - T_k, -s_0^{T_{k-1}} - T_k + T_{k-1}) & D_k = L \\ (t - s_1^{T_{k-1}} - T_k, s_0^{T_{k-1}} + T_k - T_{k-1}) & D_k = R. \end{cases} \quad (17)$$

Clearly, the number of possible destination states grows with the number of turns made. Generally, with k total number of turnings, the destination state has 2^k possibilities; when k is even, it is given by

$$\begin{bmatrix} s_0^t \\ s_1^t \end{bmatrix} = \begin{bmatrix} t - T_k \\ \pm b \end{bmatrix} + \begin{bmatrix} \pm(T_{k-1} - T_{k-2}) \\ \pm(T_k - T_{k-1}) \end{bmatrix} + \dots + \begin{bmatrix} \pm(T_1 + a) \\ \pm(T_2 - T_1) \end{bmatrix}, \quad (18)$$

and when k is odd, it is given by

$$\begin{bmatrix} s_0^t \\ s_1^t \end{bmatrix} = \begin{bmatrix} \pm b + (t - T_k) \\ \pm(T_k - T_{k-1}) \end{bmatrix} + \begin{bmatrix} \pm(T_{k-1} - T_{k-2}) \\ \pm(T_{k-2} - T_{k-3}) \end{bmatrix} + \dots + \begin{bmatrix} \pm(T_2 - T_1) \\ \pm(T_1 + a) \end{bmatrix}. \quad (19)$$

The plus/minus signs are determined by the turning directions.

By symmetry, the stationary distribution of this Markov chain is uniform with the probability $1/4n^2$. As dictated by Lemma 1, we need to find the minimum time step t such that $\Pr((s_0^t, s_1^t) = (x, y)) \geq c/n^2$ for any $(x, y) \in \{-n, \dots, -1, 1, \dots, n\}^2$ for some constant c . Intuitively, both s_0^t and s_1^t in (18) and (19) are sums of independent geometric random variables, which allows us to examine the final state probability by the Central Limit Theorem. This is done in the Lemma 3 below, which requires a technical result in Lemma 2 to simplify analysis.

Lemma 2: With high probability (w.h.p. the probability approaches 1 as $n \rightarrow \infty$), there exists a constant $c_1 > 0$ such that there are at least $\lfloor q(n)^{3/2} \rfloor$ positive odd integers (time indices) i satisfying

$$T_{i+1} - T_i \leq \lfloor n/q(n)^{3/4} \rfloor \text{ and } T_i - T_{i-1} \leq \lfloor n/q(n)^{3/4} \rfloor$$

in the first $\lfloor c_1 \sqrt{q(n)} n \rfloor$ steps.

Sketch of Proof: Since $T_{i+1} - T_i$ is a geometric random variable with parameter $2q(n)/n$, we can readily compute the probability

$$\begin{aligned} p &= \Pr(T_{i+1} - T_i \leq \lfloor n/q(n)^{3/4} \rfloor, T_i - T_{i-1} \leq \lfloor n/q(n)^{3/4} \rfloor) \\ &= \left(1 - (1 - 2q(n)/n)^{\lfloor n/q(n)^{3/4} \rfloor} \right)^2, \end{aligned} \quad (20)$$

which is non-vanishing as $n \rightarrow \infty$. By the Chernoff bound, the probability that there are at least $\lfloor q(n)^{3/2} \rfloor$ positive odd integers i with $T_{i+1} - T_i \leq \lfloor n/q(n)^{3/4} \rfloor$ and $T_i - T_{i-1} \leq \lfloor n/q(n)^{3/4} \rfloor$ in the first $c_1 q(n)^{3/2}$ steps satisfies

$$p_1 > 1 - \exp\left(-0.5 \left[1 - 1/c_1 p\right]^2 c_1 p q(n)^{3/2}\right), \quad (21)$$

which approaches 1 as $n \rightarrow \infty$, providing $c_1 > 1/p$. Furthermore, the random variable $T_{\lfloor q(n)^{3/2} \rfloor}$ is the sum of $\lfloor q(n)^{3/2}/2 \rfloor$ independent random variables with zero mean and variance between $n^2/3q(n)^2$ and $n^2/2q(n)^2$. By Chebyshev's inequality

$$\begin{aligned} \lim_{n \rightarrow \infty} \Pr\left(T_{\lfloor q(n)^{3/2} \rfloor} < c_1 \sqrt{q(n)n}\right) &= 1 - \lim_{n \rightarrow \infty} \Pr\left(T_{\lfloor q(n)^{3/2} \rfloor} \geq c_1 \sqrt{q(n)n}\right) \\ &\geq 1 - \lim_{n \rightarrow \infty} \frac{q(n)^{3/2} n^2 / q(n)^2}{2c_1^2 q(n)n^2} = 1 - \lim_{n \rightarrow \infty} \frac{1}{2c_1^2 q(n)^{3/2}} = 1. \end{aligned}$$

Thus the constant c_1 is the desired constant. \square

Divide the whole turning time index set $\{1, 2, \dots\}$ into two subsets S_1 and S_2 , where S_1 is the set of positive odd integers i such that $T_{i+1} - T_i \leq \lfloor n/q(n)^{3/4} \rfloor$ and $T_i - T_{i-1} \leq \lfloor n/q(n)^{3/4} \rfloor$, as in Lemma 2, and S_2 contains the rest. To examine the probability of the final states, it is sufficient to consider the conditional probability for any given set $T_{S_2} \sqcup \{T_j\}_{j \in S_2}$. If $i < k < j$, it is known that $P(T_i = k | T_{i-1} = i, T_{i+1} = j) = 1/(j-i-1)$. Therefore, given T_{S_2} , for every $i \in S_1$, the component vector $\begin{bmatrix} \pm(T_i - T_{i-1}) \\ \pm(T_{i+1} - T_i) \end{bmatrix}$ in (18) and (19) is a uniform random vector on the space $\Omega = \left\{ -\lfloor n/q(n)^{3/4} \rfloor, \dots, -1, 1, \dots, \lfloor n/q(n)^{3/4} \rfloor \right\}^2$. We thus can derive:

Lemma 3: There exists a constant $c > 0$ such that if $t > c_1 \sqrt{q(n)n}$,

$$\Pr\left((s'_0, s'_1) = (x, y) | T_{S_2}\right) \geq c/n^2, \quad (22)$$

for any $(x, y) \in \{-n, \dots, -1, 1, \dots, n\}^2$ and any T_{S_2} w.h.p.

Sketch of Proof: From Lemma 2, we know that if $t > c_1 \sqrt{q(n)n}$, there are at least $\lfloor q(n)^{3/2} \rfloor$ uniform random vectors on Ω , with zero mean and covariance matrix Σ whose entries are all on the order of $n^2/q(n)^{3/2}$. Denote these random vectors as $X_m(n)$, $m = 1, \dots, \lfloor q(n)^{3/2} \rfloor$ and let $Y_m(n) = \Sigma^{-1/2} X_m(n) / q(n)^{3/4}$, then from the Central Limit Theorem, the summation of $Y_m(n)$ converges to a standard multivariate normal distribution

$$\mathbf{S}_n = \sum_{m=1}^{\lfloor q(n)^{3/2} \rfloor} Y_m(n) \sim \mathcal{N}(\mathbf{0}_{2 \times 1}, \mathbf{I}_{2 \times 2}). \quad (23)$$

Let $c = \Pr(1 \leq \|\mathbf{S}_n\| \leq 2)$, where $\|\cdot\|$ denotes the Euclidean distance, then

$$\Pr\left(n \leq \left\| q(n)^{3/4} \Sigma^{1/2} \mathbf{S}_n \right\| \leq 2n\right) \geq c. \quad (24)$$

From above, we know \mathbf{S}_n is a zero-mean unimodal symmetric random vector, therefore

$$\Pr\left(\left\| q(n)^{3/4} \Sigma^{1/2} \mathbf{S}_n \right\| = n\right) \geq c/n^2. \quad (25)$$

Let U_n be the sum of $\lfloor q(n)^{3/2} \rfloor$ i.i.d. random vectors which are uniform on Ω , then U_n and $q(n)^{3/4} \Sigma^{1/2} \mathbf{S}_n$ have the same distribution. Thus if $t > c_1 \sqrt{q(n)n}$

$$\Pr\left(\begin{bmatrix} s'_0 & s'_1 \end{bmatrix}^T = [x, y]^T\right) \geq c/n^2,$$

for all $[x, y]^T \in \{-n, \dots, -1, 1, \dots, n\}^2$. \square

Combining Lemma 1 and 3, we get the following conclusion:

Theorem 1: On a 2-d $n \times n$ torus, the mixing time of an edge process with turning probability $q(n)/n$ is $O(\sqrt{q(n)n})$ w.h.p.

As a result, the convergence time of consensus propagation [2] and our intra-cluster BP inference (6) is upper bounded by $O(n^{3/2})$, with $q(n)/n = c$ for some constant c in the *worst* case.

IV. PERFORMANCE-COMPLEXITY ANALYSIS FOR STRUCTURED VARIATIONAL METHODS

It has been shown in Section II.B that the performance of structured variational methods is governed by a mixed vertex-edge process, or equivalently a hybrid Markov chain model. The complexity of this model precludes *direct* applications of any standard techniques in literature. In this section, we explore the coupling technique [9] to analyze this model.

Coupling provides a simple and elegant way of bounding the mixing time, and isn't tied to reversibility. Essentially, a coupling of Markov chains is a process $\{X_t, Y_t\}_{t=0}^\infty$ with the property that both $\{X_t\}$ and $\{Y_t\}$ are Markov chains with the same transition matrix \hat{P} of interest, but typically with different starting states. Once the two chains simultaneous visit to a single state, they stay together at all times after that, i.e.

$$\text{If } X_{t'} = Y_{t'}, \text{ then } X_t = Y_t \text{ for } t \geq t'.$$

For starting states x and y , let

$$T^{x,y} = \min\{t : X_t = Y_t | X_0 = x, Y_0 = y\}, \quad (26)$$

then the coupling time is defined as

$$t_{\text{couple}} = \max_{x,y} E(T^{x,y}), \quad (27)$$

which can serve as an upper bound for the mixing time according to the Coupling Lemma [9]:

$$t_{\text{mix}}(\varepsilon) \leq t_{\text{couple}} \ln \varepsilon. \quad (28)$$

We assume an $n \times n$ torus is equally divided into $s \times s$ clusters, each of size $(n/s) \times (n/s)$, and consider a vertex-edge process on it as indicated in Fig.1 (c). Then by investigating the coupling time of two random walks on such a clustered graph, we can obtain a characterization for the mixing time. Firstly, we study how quickly an edge process can "escape" from a 2-d torus (or how long it can stay in the torus before hitting any outgoing edges on the boundaries), and obtain the following result:

Lemma 4: On a 2-d $n \times n$ torus, the average staying time of an edge process is upper-bounded by $O(\sqrt{q(n)n})$ w.h.p.

Sketch of Proof: From the state representation of edge process (c.f. Fig. 2), we know hitting an outgoing edge on the boundary corresponds to $s'_0 = -1$ or n . According to (18) and (19), s'_0 is given by

$$s'_0 = \begin{cases} t - T_k \pm (T_{k-1} - T_{k-2}) + \dots + \pm(T_1 + a) & k \text{ is even} \\ \pm b + (t - T_k) \pm (T_{k-1} - T_{k-2}) + \dots + \pm(T_2 - T_1) & k \text{ is odd} \end{cases},$$

which is the sum of $k = O(tq(n)/n)$ uniformly distributed random variables with zero mean and variance $n^2/q(n)^{3/2}$. When $t = c_2 \sqrt{q(n)n}$ (i.e. $k = O(q(n)^{3/2})$) and only consider $s'_0 = n$, we can use local limit theorem ([10], page 10) to get

$$\frac{n}{q(n)^{3/4}} q(n)^{3/4} \Pr(s'_0 = n) \rightarrow \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{n^2}{2(n^2/q(n)^{3/2})q(n)^{3/2}}\right\}, \quad (29)$$

or $\Pr(s'_0 = n) \rightarrow 1/n\sqrt{2\pi e}$ as $n \rightarrow \infty$. Then by the Chernoff bound, the probability that there is at least one time that the walk hits the boundary edges before $t = c_2 \sqrt{q(n)n}$ is

$$p_2 > 1 - \exp\left(-0.5\left[1 - 1/c_3\sqrt{q(n)}\right]^2 c_3\sqrt{q(n)}\right) \rightarrow 1, \quad (30)$$

where $c_3 = c_2/\sqrt{2\pi e}$. So $c_2\sqrt{q(n)}n$ is an upper bound for the average staying time. \square

Using this result, the mixing time of a mixed vertex-edge process can be characterized as follows:

Theorem 2: On a 2-d $n \times n$ torus, the mixing time of a mixed vertex-edge process with equal cluster size of $(n/s) \times (n/s)$ is

$$O\left(\sqrt{sn^{3/2}}\right) \text{ w.h.p.}$$

Sketch of Proof: Suppose two random walks start from two randomly selected points in the clustered graph, then the coupling process can be described as follows: firstly, these two random walks wander inside their respective clusters (edge process) until they hit the gateway nodes and exit the starting clusters. From then on, they roam over the network, repeatedly entering and leaving clusters, and finally arrive at a same cluster. This journey, on the high level, can be regarded as a vertex process on an $s \times s$ torus with ‘‘mega-vertices’’, which take $O(s^2)$ steps to couple. At each mega-vertex (cluster), the average staying time is $O((n/s)^{3/2})$ according to Lemma 4. Besides, we need to consider the scenario that even these two walks reach the same cluster; one of them may leave early, by hitting gateway nodes before coupling with the other walk. In this case, the above journey is repeated. We assume this probability as $p_\tau = \Pr(\tau_{\text{hit}} < \tau_{\text{couple}})$, where τ_{hit} and τ_{couple} are the expected time to hit a boundary node in a cluster and the coupling time in the same cluster, respectively. Then the total time to couple these two random walks is upper bounded by

$$\begin{aligned} t_{\text{couple}} &\leq \sum_{i=1}^{\infty} i \left[O(s^2)O((n/s)^{3/2}) + O((n/s)^{3/2}) \right] (1-p_\tau) p_\tau^{i-1} \\ &= O(s^2)O((n/s)^{3/2})/(1-p_\tau) + O((n/s)^{3/2})/(1-p_\tau), \end{aligned} \quad (31)$$

where the first term gives the total roaming time among the clusters, while the second term corresponds to the staying time or coupling time in the same cluster of two random walks.

To evaluate p_τ , assume x is the state of the random walk which just steps into a cluster, z is a boundary node of that cluster, and a is the (dynamic) state of the random walk which has entered this cluster earlier. From [11], the probability that starting from node x , a random walk hits node z before it hits a is given by the ratio of the effective resistance² between a and x and that between a and z :

$$p_\tau = P_x(\tau_{\text{hit},z} < \tau_{\text{hit},a}) = R(a \leftrightarrow x) / R(a \leftrightarrow z). \quad (32)$$

Since both x and z are boundary states, while we don't have any further information about a but to assume it is uniformly randomly located inside the cluster, $R(a \leftrightarrow x)$ and $R(a \leftrightarrow z)$ are on the same order, and so p_τ is a constant.

We thus have

$$t_{\text{mix}}(\varepsilon) \leq t_{\text{couple}} \ln \varepsilon = O(s^2(n/s)^{3/2}) = O\left(\sqrt{sn^{3/2}}\right). \quad (33)$$

\square

Note that Theorem 2 includes previous results for the vertex process ($s = n$) and edge process ($s = 1$) as two special cases.

To inspect how clustering affects the message complexity, note for the total s^2 clusters, each cluster has $(n/s)^2$ nodes and hereby $4(n/s)^2$ directed edges. On each directed edge, two metrics, the message mean and variance, are exchanged in the BP algorithm, except for the $4(n/s)$ outgoing ones across the cluster boundaries. Instead, on these cross-cluster edges, only the estimated means are transmitted. So the message complexity per iteration in the SMF algorithm is

$$O\left(s^2 \left\{ 2 \left[4(n/s)^2 - 4(n/s) \right] + s^2 4(n/s) \right\}\right) = O(8n^2 - 4ns). \quad (34)$$

Comparing (33) and (34), we can observe the performance-complexity tradeoff inherent in SMF: as the cluster size increases, accurate algorithms are performed on more nodes; this results in faster convergence, but also inevitably increases communication burden. An appropriate cluster size should be selected for SMF depending on applications, to achieve a balance among estimation accuracy, convergence rate, computation complexity, and energy efficiency.

SMF also requires some overhead for clustering, which can be done before the network setup and can be adjusted during network operation when necessary. We have designed a distributed clustering scheme for SMF, which can minimize the dependence between the clusters, thus improve the algorithm performance. Due to space limitation, this clustering scheme along with some numerical results will be reported in our later publications.

V. CONCLUSIONS

In this paper, we investigate the asymptotic performance of our recently proposed structured variational methods for distributed inference. We adopt a direct probabilistic approach to analyze the convergence of an edge process which models the intra-cluster processing, and devise a coupling process to characterize the overall performance concerning a more complicated mixed vertex-edge process. The tradeoff between the complexity and performance in this algorithm is also addressed. We expect both the results obtained and the analytical tools developed in this work can be applied to other similar problems in wireless networks.

REFERENCES

- [1] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA: Morgan Kaufmann, 1988.
- [2] C. C. Moallemi and B. Van Roy, ‘‘Consensus Propagation,’’ *IEEE Transactions on Information Theory*, Vol. 52, No. 11, pp. 4753–4766, 2006.
- [3] S. Boyd, A. Ghosh, B. Prabhakar and D. Shah, ‘‘Randomized gossip algorithms,’’ *IEEE Trans. Inform. Theory*, vol. 52, no. 6, June 2006.
- [4] Y. Zhang and H. Dai, ‘‘Structured Variational Methods for Distributed Inferences in Wireless Ad-hoc and Sensor Networks,’’ *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'09)*, Taipei, Apr. 2009.
- [5] P. Diaconis, S. Holmes, and R. M. Neal, ‘‘Analysis of a non-reversible markov chain sampler,’’ Biometrics Unit, Cornell University, Tech. Rep. BU-1385-M, 1997.
- [6] M. Hildebrand, ‘‘Rates of Convergence of the Diaconis-Holmes-Neal Markov Chain Sampler,’’ preprint.
- [7] W. Li and H. Dai, ‘‘Accelerating distributed consensus via lifting Markov chains,’’ *2007 IEEE International Symposium on Information Theory (ISIT)*, Nice, France, June 2007.
- [8] D. Randall, ‘‘Rapidly mixing Markov chains with applications in computer science and physics,’’ *Computing in Science and Engineering*, Vol. 8, No 2, March 2006.
- [9] T. Lindvall, ‘‘Lectures on the Coupling Method,’’ Courier Dover Publications, 2002.
- [10] V. F. Kolchin, ‘‘Random Graph,’’ Cambridge University Press, 1999.
- [11] D. A. Levin, Y. Peres and E. L. Wilmer, ‘‘Markov Chains and Mixing Times,’’ American Mathematical Society, 2008.
- [12] S. Boyd, A. Ghosh, B. Prabhakar and D. Shah, ‘‘Mixing Times for Random Walks on Geometric Random Graphs,’’ *SIAM Workshop on Analytic Algorithms & Combinatorics (ANALCO)*, Vancouver, Canada, January 2005.

² The effective resistance between node i and j is the expected number of traversals in a random walk starting at i and ending in j [11].