Stein Variational Message Passing for Continuous Graphical Models

Dilin Wang* [1]   **Zhe Zeng*** [2]   **Qiang Liu** [1]

Abstract

We propose a novel distributed inference algorithm for continuous graphical models, by extending Stein variational gradient descent (SVGD) (Liu & Wang, 2016) to leverage the Markov dependency structure of the distribution of interest. Our approach combines SVGD with a set of structured local kernel functions defined on the Markov blanket of each node, which alleviates the curse of high dimensionality and simultaneously yields a distributed algorithm for decentralized inference tasks. We justify our method with theoretical analysis and show that the use of local kernels can be viewed as a new type of *localized* approximation that matches the target distribution on the conditional distributions of each node over its Markov blanket. Our empirical results show that our method outperforms a variety of baselines including standard MCMC and particle message passing methods.

1. Introduction

Probabilistic graphical models, such as Markov random fields (MRFs) and Bayesian networks, provide a powerful framework for representing complex stochastic dependency structures between a large number of random effects (Pearl, 1988; Lauritzen, 1996). A key challenge, however, is to develop computationally efficient inference algorithms to approximate important integral quantities related to distributions of interest. Variational message passing methods, notably belief propagation (Pearl, 1988; Yedidia et al., 2003), provide one of the most powerful frameworks for approximate inference in graphical models (Wainwright & Jordan, 2008). In addition to accurate approximation, message passing algorithms perform inference in a distributed fashion by passing messages between variable nodes to

---

*Equal contribution [1]Department of Computer Science, The University of Texas at Austin [2]School of Mathematical Sciences, Zhejiang University. Correspondence to: Dilin Wang <dilin@cs.utexas.edu>, Qiang Liu <lqiang@cs.utexas.edu>.

propagate uncertainty, well suited for *decentralized inference* tasks such as sensor network localization (e.g., Ihler et al., 2005).

Unfortunately, standard belief propagation (BP) is best applicable only to discrete or Gaussian variable models. Significant additional challenges arise when applying BP to continuous, non-Gaussian graphical models, because the exact BP updates involve intractable integration. As a result, the existing continuous variants of BP require additional particles or non-parametric approximation (e.g., Ihler et al., 2005; Ihler & McAllester, 2009; Sudderth et al., 2010; Song et al., 2011), which deteriorates the accuracy and stability. Another key aspect that was not well discussed in the literature is that the traditional continuous BP methods are gradient-free, in that they do not use the gradient of the density function. Although this makes the methods widely applicable, their performance can be significantly improved by incorporating the gradient information whenever available.

Stein variational gradient descent (SVGD) (Liu & Wang, 2016) is a recent particle-based variational inference algorithm that combines the advantages of variational inference and particle-based methods, and efficiently leverages the gradient information for continuous inference. Unlike traditional variational inference that constructs parametric approximation of the target distribution by minimizing KL divergence, SVGD directly approximates the target distribution with a set of particles, which is iteratively updated following a velocity field that decreases the KL divergence with the fastest speed among all possible velocity fields in a reproducing kernel Hilbert space (RKHS) of a positive definite kernel. This makes SVGD inherit the theoretical consistency of particle methods (Liu, 2017), while obtaining the fast practical convergence thanks to its deterministic updates. The goal of this work is to adapt SVGD for distributed inference of continuous graphical models.

In principle, one can directly apply SVGD to continuous graphical models. However, standard SVGD does not yield a distributed message passing like BP, because its update involves a kernel function, which is defined on all the variable dimensions, introducing additional dependency beyond the Markov blanket of the graphical model of interest. In addition, the use of the global kernel function on all the

variables also deteriorates the performance in high dimensions; our empirical findings show that although SVGD tends to perform exceptionally well in estimating the mean parameters, it becomes less sample efficient in terms of estimating the variances as the dimension increases.

In this paper, we improve SVGD to take advantage of the inherent dependency structure of graphical models for better distributed inference. Instead of using a global kernel function, we associate each node with a local kernel function that only depends on the Markov blanket of each node. This simple modification allows us to turn SVGD into a distributed message passing algorithm, and simultaneously alleviates the curse of dimensionality.

Theoretically, our method extends the original SVGD in two significant ways: *i*) it uses different kernel functions for different coordinates (or variable nodes), justified with a theoretical analysis that extends the results of Liu et al. (2016); Liu & Wang (2016); *ii*) it uses a local kernel over the Markov blanket for each node, which, compared with the typical global kernel function, can be viewed as introducing a type of deterministic approximation to trade for better sample efficiency. Our empirical results show that our method outperforms a variety of baseline methods, including the typical SVGD and Monte Carlo, and particle message passing (PMP). We note that a similar idea is independently and concurrently proposed by Zhuo et al. (2018) in the same conference proceeding.

2. Background

We introduce the background of Stein variational gradient descent (SVGD) and kernelized Stein discrepancy (KSD), which forms the foundation of our work. For notation, we denote by $x = [x_1, \ldots, x_d]$ a vector in $\mathbb{R}^d$ and $\{x^\ell\}_{\ell=1}^n$ a set of n vectors.

Stein Variational Gradient Descent (SVGD) Let $p(x)$ be a positive differentiable density function on $\mathbb{R}^d$. Our goal is to find a set of points (or particles) $\{x^\ell\}_{\ell=1}^n$ to approximate p so that $\sum_{\ell=1}^n f(x^\ell)/n \approx \mathbb{E}_p[f]$ for general test functions f. SVGD achieves this by iteratively updating the particles with deterministic transforms of form

$$x^\ell \leftarrow x^\ell + \epsilon \phi(x^\ell), \qquad \forall \ell = 1, \ldots, n,$$

where ϵ is a small step size and ϕ is a velocity field that decides the update directions of the particles. The key problem is to choose an optimal velocity field ϕ to decrease the KL divergence between the distribution of particles and the target $p(x)$ as fast as possible. This can be solved by the following basic observation shown in Liu & Wang (2016): assume $x \sim q$ and $q_{[\epsilon\phi]}$ is the distribution of $x' = x + \epsilon \phi(x)$, then

$$\mathrm{KL}(q_{[\epsilon\phi]} \| p) = \mathrm{KL}(q \| p) - \epsilon \, \mathbb{E}_{x \sim q}[\mathcal{P}_x^\top \phi(x)] + O(\epsilon^2),$$

where $\mathcal{P}_x$ is a linear operator, called Stein operator, that acts on function ϕ via

$$\mathcal{P}_x^\top \phi(x) \overset{\text{def}}{=} \nabla_x \log p(x)^\top \phi(x) + \nabla_x^\top \phi(x). \quad (1)$$

This result suggests that the decreasing rate of KL divergence when applying transform $x' = x + \epsilon\phi(x)$ is dominated by $\mathbb{E}_{x \sim q}[\mathcal{P}_x^\top \phi(x)]$ when the step size ϵ is small. In a special case when $q = p$, Stein operator draws connection to Stein's identity, which shows

$$\mathbb{E}_{x \sim q}[\mathcal{P}_x^\top \phi(x)] = 0 \quad \text{when} \quad p = q.$$

This is expected because as $q = p$, the decreasing rate of KL divergence must be zero for all velocity fields ϕ.

Given a candidate set $\mathcal{F}$ of velocity fields ϕ, we should choose the best ϕ to maximize the decreasing rate,

$$\mathbb{D}(q \| p) \overset{\text{def}}{=} \max_{\phi \in \mathcal{F}} \left\{ \mathbb{E}_{x \sim q}[\mathcal{P}_x^\top \phi(x)] \right\}, \quad (2)$$

where the maximum decreasing rate $\mathbb{D}(q \| p)$ is called the *Stein discrepancy* between q and p. Assume $\mathcal{F}$ includes $\pm\phi$ for $\forall \phi \in \mathcal{F}$, then (2) is equivalent to maximizing the absolute value of $\mathbb{E}_q[\mathcal{P}_x^\top \phi]$ and $\mathbb{D}(q \| p)$ must be non-negative for any q and p. In addition, If $\mathcal{F}$ is taken to be rich enough, $\mathbb{D}(q \| p) = 0$ only if there exists no velocity field ϕ that can decrease the KL divergence between p and q, which must imply $p = q$.

To make $\mathcal{F}$ computationally tractable, Liu & Wang (2016) further assumed that $\mathcal{F}$ is the unit ball of a vector-valued RKHS $\mathcal{H} = \mathcal{H}_0 \times \cdots \times \mathcal{H}_0$, where each $\mathcal{H}_0$ is a scalar-valued RKHS associated with a positive definite kernel $k(x, x')$. In this case, Liu et al. (2016) showed that the optimal solution of (2) is $\phi^* / ||\phi^*||$, where

$$\phi^*(\cdot) = \mathbb{E}_{x \sim q}[\mathcal{P}_x k(x, \cdot)]$$
$$= \mathbb{E}_{x \sim q}[\nabla_x \log p(x) k(x, \cdot) + \nabla_x k(x, \cdot)]. \quad (3)$$

This gives the optimal update direction within RKHS $\mathcal{H}$. By starting with a set of initial particles and then repeatedly applying this update with q replaced by the empirical distributions of the particles, we obtain the SVGD algorithm:

$$x^\ell \leftarrow x^\ell + \epsilon \phi^*(x^\ell), \qquad \forall \ell = 1, \ldots, n, \quad (4)$$

$$\phi^*(\cdot) = \frac{1}{n} \sum_{\ell=1}^n [\nabla_{x^\ell} \log p(x^\ell) k(x^\ell, \cdot) + \nabla_{x^\ell} k(x^\ell, \cdot)].$$

The two terms in $\phi^*(x)$ play different roles: the first term with the gradient $\nabla_x \log p(x)$ drives the particles toward the high probability regions of $p(x)$, while the second term with $\nabla_{x^\ell} k(x, x^\ell)$ serves as a repulsive force to encourage diversity as shown in Liu & Wang (2016). It is easy to see from (4) that SVGD reduces to standard gradient ascent for maximizing $\log p(x)$ (i.e., maximum a posteriori (MAP)) when there is only a single particle ($n = 1$).

Discriminative Stein Discrepancy   One key requirement when selecting the function space $\mathcal{H}$ is that it should be rich enough to make Stein discrepancy discriminative, that is,

$$\mathbb{D}(q \,\|\, p) = 0 \qquad \text{implies} \qquad q = p. \qquad (5)$$

This problem has been studied in various recent works, including Liu et al. (2016); Chwialkowski et al. (2016); Oates et al. (2016); Gorham & Mackey (2017), all of which require $\mathcal{H}$ to be an universal approximator in certain sense. In particular, the condition required in Liu et al. (2016) is that the kernel $k(x, x')$ should be *strictly integrally positive definite* in the sense that

$$\int g(x)k(x, x')g(x')dxdx' > 0, \qquad (6)$$

for any nonzero function g with $0 < \|g\|_2^2 < \infty$. Many widely used kernels, including Gaussian RBF kernels of form $k(x, x') = \exp(-\frac{1}{h}\|x - x'\|_2^2)$, are strictly integrally positive definite.

3. SVGD for Graphical Models

The goal of our work is to extend SVGD to approximate high dimensional probabilistic graphical models of form

$$p(x) \propto \exp\Big[\sum_{s \in \mathcal{S}} \psi(x_s)\Big], \qquad (7)$$

where $\mathcal{S}$ is a family of index sets $s \subseteq \{1, \ldots, d\}$, and $x_s = [x_i]_{i \in s}$ represents the sub-vector of x over index set s. Here the clique set $\mathcal{S}$ specifies the *Markov structure* of the graphical model: for each variable (or node) i, its Markov blanket (or neighborhood) $\mathcal{N}_i$ is the set of nodes that co-appears in at least one clique $s \in \mathcal{S}$, that is, $\mathcal{N}_i := \cup\{s\colon s \in \mathcal{S}, \ s \ni i\} \setminus \{i\}$. Related, we define the *closed* neighborhood (or clique) of node i to be $\mathcal{C}_i := \mathcal{N}_i \cup \{i\}$. The local Markov property guarantees that a variable x_i is conditionally independent of all other nodes given its Markov blanket $\mathcal{N}_i$.

The Markov structure is also reflected in gradient $\nabla \log p$:

$$\partial_{x_i} \log p(x) = \sum_{s \ni i} \partial_{x_i} \psi(x_s),$$

where ∂_{x_i} denotes the partial derivative with respect to x_i, the i-th component of x. This suggests that the gradient evaluation of node i only requires information from its closed neighborhood set $\mathcal{C}_i$; this property makes gradient-based distributed inference methods a possibility.

Unfortunately, when directly applying SVGD to graphical models, the Markov structure is not inherited in the SVGD update (4), because typical kernel functions do not have the same (additive) factorization structure as $\partial_{x_i} \log p(x)$.
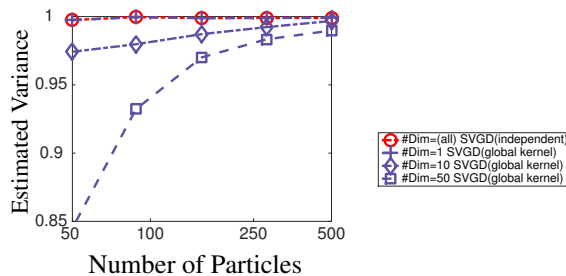

Figure 1. Estimating the variance using SVGD when $p(x)$ is the standard multivariate Gaussian distribution $\mathcal{N}(0, I_d)$ of different dimensions d (the true variance is $\sigma = 1$).

Take the commonly used Gaussian RBF kernel $k(x, x') = \exp(-\frac{1}{h}\|x - x'\|_2^2)$ as an example. Because $k(x, x')$ involves all the coordinates of x, the i-th coordinate of the Stein variational gradient $\phi^*(x)$ in (4) depends on all the other nodes, including those out of the Markov blanket of node i. This makes it infeasible to apply vanilla SVGD in distributed settings because the update of node i requires information from all the other nodes.

More importantly, this global dependency introduced by kernel functions may also lead to poor performance in high dimensional models. To illustrate this, we consider a simple example when the distribution $p(x)$ of interest is fully factorized, that is, $p(x) = \prod_{i=1}^d p_i(x_i)$, where each node x_i is independent of all the other nodes. In this case, we find that SVGD with the standard Gaussian RBF kernel (see *SVGD (global kernel)* in Figure 1) requires increasingly more particles in order to estimate the variance accurately as the dimension d increases; this is caused by the additional (and unnecessary) dependencies introduced by the use of the global RBF kernel.

In this particular case, an obviously better approach is to apply SVGD with RBF kernel on each of the one-dimensional marginal $p_i(x_i)$ individually (see *SVGD (independent)* in Figure 1)); this naturally leverages the fully factorized structure of $p(x)$, and makes the algorithm immune to the curse of dimensionality because the RBF kernel is applied on an individual variable each time. Therefore, exploiting the sparse dependency structure can significantly improve the performance in high dimensions. The key question, however, is how we can extend the idea beyond the fully factorized case. This motivates our graphical SVGD algorithm as we show in the sequel.

3.1. Graphical SVGD

In order to extend the above example, we observe that running SVGD on each marginal distribution $p_i(x_i)$ independently can be viewed as a special SVGD applied on the joint distribution $p(x) = \prod_i p_i(x_i)$, but updating each co-

ordinate x_i using its own kernel $k_i(x, x')$ that only depends on the i-th coordinate, that is, $k_i(x, x') := k_i(x_i, x'_i)$. An intuitive way to extend this to $p(x)$ with more general Markov structures is to update each x_i with a local kernel function $k_i(x, x') := k_i(x_{\mathcal{C}_i}, x'_{\mathcal{C}_i})$ that depends only on the closed neighborhood $\mathcal{C}_i$ of node i, that is,

$$x^\ell_i \leftarrow x^\ell_i + \epsilon \phi^*_i(x^\ell), \qquad \forall i \in [d], \ \ell \in [n] \qquad (8)$$

$$\phi^*_i(x) := \frac{1}{n} \sum_{\ell=1}^{n} [s_i(x^\ell) k_i(x^\ell_{\mathcal{C}_i}, x_{\mathcal{C}_i}) + \partial_{x^\ell_i} k_i(x^\ell_{\mathcal{C}_i}, x_{\mathcal{C}_i})],$$

where $s_i(x) = \partial_{x_i} \log p(x)$. This procedure, which we call graphical SVGD (see also Algorithm 1), provides a simple way to efficiently integrate the graphical structure of $p(x)$ into SVGD. As we demonstrate in our experiments, it significantly improves the performance in high dimensional, sparse graphical models. In addition, it yields a communication-efficient distributed message passing form, since the update of x_i only requires to access the neighborhood variables in $\mathcal{C}_i$.

Is this graphical SVGD update theoretically sound? The key difference between (8) and the original SVGD (4) includes: i) graphical SVGD uses a different kernel $k_i(x, x')$ for each coordinate x_i; and ii) each kernel $k_i(x, x') = k_i(x_{\mathcal{C}_i}, x'_{\mathcal{C}_i})$ only depends on the closed neighborhood $\mathcal{C}_i = \{i\} \cup \mathcal{N}_i$ of node i. We justify these two choices theoretically in Section 3.2 and Section 3.3, respectively.

3.2. Stein Discrepancy with Coordinate-wise Kernels

Using coordinate-wise kernel $k_i(x, x')$ can be simply viewed as taking the space $\mathcal{H}$ in the functional optimization (2) to be a more general product space $\mathcal{H} = \mathcal{H}_1 \times \cdots \times \mathcal{H}_d$, where each individual RKHS $\mathcal{H}_i$ is related to a different kernel $k_i(x, x')$; the original SVGD can be viewed as a special case of this when all the kernels $k_i(x, x')$, $\forall i \in [d]$ equal. The following result extends Lemma 3.2 and Theorem 3.3 of Liu & Wang (2016) to take into account coordinate-wise kernels.

Theorem 1. *Let $\mathcal{H}_i$ be the (scalar-valued) RKHS related to kernel $k_i(x, x')$, and $\mathcal{H} = \mathcal{H}_1 \times \cdots \times \mathcal{H}_d$ their product RKHS consisting of $\phi = [\phi_1, \ldots, \phi_d]^\top$ with norm $\|\phi\|^2_{\mathcal{H}} = \sum_i \|\phi_i\|^2_{\mathcal{H}_i}$, $\forall \phi_i \in \mathcal{H}_i$. Taking $\mathcal{F} = \{\phi \in \mathcal{H} : \|\phi\|_{\mathcal{H}} \leq 1\}$ in the optimization problem in (2), then the optimal solution is*

$$\phi^*(\cdot) = \mathbb{E}_{x \sim q}[\mathcal{P}_x \circ \boldsymbol{k}(x, \cdot)], \qquad (9)$$

with $\mathcal{P}_x \circ \boldsymbol{k}(x, \cdot) \overset{def}{=} [\mathcal{P}_{x_1} k_1(x, \cdot), \cdots, \mathcal{P}_{x_d} k_d(x, \cdot)]^\top$,

where $\circ$ denotes the entrywise product between $\mathcal{P}_x$ and $\boldsymbol{k} = [k_1, \ldots, k_d]$. Further, the related Stein discrepancy in (2) equals $\mathbb{D}(q \mid\mid p) = \|\phi^*\|_{\mathcal{H}}$.

Algorithm 1 Graphical Stein Variational Gradient Descent

Input: A graphical model $p(x)$ with Markov blanket $\mathcal{N}_i$ for node i; $\mathcal{C}_i = \mathcal{N}_i \cup \{i\}$; a set of local kernels $k_i(x_{\mathcal{C}_i}, x'_{\mathcal{C}_i})$, and initial particles $\{x^{\ell,0}\}_{\ell=1}^n$; step size ϵ.
Goal: A set of particles $\{x^\ell\}_{\ell=1}^n$ that approximates $p(x)$
for iteration t **do**
  for node i **do**

$$x^{\ell,t+1}_i \leftarrow x^{\ell,t}_i + \epsilon \phi^*_i(x^{\ell,t})$$

    where $\phi^*_i(x)$ is defined in (8) (with $x^\ell = x^{\ell,t}$).
  end for
end for

This result suggests that updates of form $x'_i \leftarrow x'_i + \epsilon \mathbb{E}_{x \sim q}[\mathcal{P}_{x_i} k_i(x, x')]$ yields the fastest descent direction of KL divergence within $\mathcal{H}$, justifying the graphical SVGD update in (8).

The following result studies the properties of the Stein discrepancy related to coordinate-wise kernels $\boldsymbol{k} = [k_1, \ldots, k_d]$, showing that $\mathbb{D}(q \mid\mid p)$ is discriminative if all the kernels $k_i(x, x')$ are strictly integrally positive definite, a result that generalizes Proposition 3.3 of Liu et al. (2016).

Theorem 2. *Following Theorem 1, the Stein discrepancy $\mathbb{D}(q \mid\mid p)$ with kernels $\boldsymbol{k} = [k_1, \ldots, k_d]$ satisfies*

$$\mathbb{D}(q \mid\mid p)^2 = \sum_{i=1}^{d} \mathbb{E}_{x, x' \sim q}[\mathcal{P}_{x_i} \mathcal{P}_{x'_i} k_i(x, x')].$$

Further, Assume both $p(x)$ and $q(x)$ are positive and differentiable densities. Denote by $\mathcal{Q}_x$ the Stein operator of distribution q. If Stein's identity $\mathbb{E}_{x \sim q}[\mathcal{Q}_{x_i} k_i(x, x')] = 0$, $\forall x' \in \mathcal{X}$ holds for all the kernels k_i, $\forall i \in [d]$, we have

$$\mathbb{D}(q \mid\mid p)^2 = \sum_{i=1}^{d} \mathbb{E}_{x, x' \sim q}[\delta_i(x) k_i(x, x') \delta_i(x')], \qquad (10)$$

where $\delta_i(x) = \nabla_{x_i} \log p(x_i | x_{\neg i}) - \nabla_{x_i} \log q(x_i | x_{\neg i})$, with $\neg i = \{1, \ldots, n\} \setminus \{i\}$.

Assume $\|q \delta_i\|^2_2 < \infty$, $\forall i \in [d]$. If all the kernels $k_i(x, x')$ are strictly integrally positive definite in the sense of (6), then $\mathbb{D}(q \mid\mid p) = 0$ implies $q = p$.

3.3. Stein Discrepancy with Local Kernels

Theorem 2 requires every kernel $k_i(x, x')$ to be strictly integral positive definite to make Stein discrepancy discriminative. However, in our graphical SVGD, each kernel function $k_i(x, x') := k_i(x_{\mathcal{C}_i}, x'_{\mathcal{C}_i})$ only depends on a subset of the variables and can be easily seen to be not strictly integrally positive definite in the sense of (6). As a result, the related Stein discrepancy is no longer discriminative in general.

Fortunately, as we show in the following, once each $k_i(x_{\mathcal{C}_i}, x'_{\mathcal{C}_i})$ is strictly integrally positive definite w.r.t. its own local variable domain $x_{\mathcal{C}_i}$, that is,

$$\int g(x_{\mathcal{C}_i}) k_i(x_{\mathcal{C}_i}, x'_{\mathcal{C}_i}) g(x'_{\mathcal{C}_i}) dx_{\mathcal{C}_i} dx'_{\mathcal{C}_i} > 0 \qquad (11)$$

for any function $g(x_{\mathcal{C}_i})$ with $0 < ||g||_2^2 < \infty$, then a zero Stein discrepancy $\mathbb{D}(q \,||\, p) = 0$ guarantees to match the conditional distributions:

$$q(x_i \,|\, x_{\mathcal{N}_i}) = p(x_i \,|\, x_{\mathcal{N}_i}) \quad \text{for any } i \in [d].$$

Although this in general does not necessarily imply that the joint distribution equals $(q = p)$ (unless q is guaranteed a priori to have the same Markov structure as p), it suggests that graphical SVGD captures important perspectives of the target distribution, especially in terms of the quantities related to the local dependency structures.

Theorem 3. *Assuming that $p(x)$ is a graphical model in which the Markov neighborhood of node i is $\mathcal{N}_i$ and $\mathcal{C}_i = \mathcal{N}_i \cup \{i\}$, and that $k_i(x, x') = k_i(x_{\mathcal{C}_i}, x'_{\mathcal{C}_i})$, $\forall i \in [d]$, then* (10) *reduces to*

$$\mathbb{D}(q \,||\, p)^2 = \sum_{i=1}^d \mathbb{E}_{x,x' \sim q}[\delta_i(x_{\mathcal{C}_i}) k_i(x_{\mathcal{C}_i}, x'_{\mathcal{C}_i}) \delta_i(x'_{\mathcal{C}_i})],$$

where $\delta_i(x_{\mathcal{C}_i}) = \nabla_{x_i} \log p(x_i|x_{\mathcal{N}_i}) - \nabla_{x_i} \log q(x_i|x_{\mathcal{N}_i})$. Further, assume $g(x_{\mathcal{C}_i}) := q(x_{\mathcal{C}_i}) \delta_i(x_{\mathcal{C}_i})$ and $||g||_2^2 < \infty$. If each kernel $k_i(x_{\mathcal{C}_i}, x_{\mathcal{C}_i})$ is integrally strictly positive definite w.r.t. variable $x_{\mathcal{C}_i}$ as defined in (11)*, we have*

$$\mathbb{D}(q \,||\, p) = 0 \quad \textit{iff} \quad q(x_i|x_{\mathcal{N}_i}) = p(x_i|x_{\mathcal{N}_i}), \ \forall i \in [d].$$

Generally speaking, matching the condition distributions shown in Theorem 3 does not guarantee to match the joint distributions $(p = q)$. A simple counter example is when $p(x)$ is fully factorized, $p(x) = \prod_i p_i(x_i)$ and $\mathcal{N}_i = \emptyset$, in which case matching the singleton marginals $p(x_i) = q(x_i)$, $i \in [d]$ does not imply $p(x) = q(x)$ jointly since one can construct infinite many distributions with the same singleton marginals using the Copula method (Nelsen, 2007).

Therefore, graphical SVGD can be viewed as a *partial* reconstruction of the target distribution p, which retains the local dependency structures while ignoring the long-range relations which are inherently difficult to infer due to the curse of high dimensionality. Focusing on the local dependency structures makes the problem easier and tractable, and also of practical interest since local marginals are often what we care in practice. Therefore, graphical SVGD can be viewed as an interesting hybrid of deterministic approximation (via the use of local kernels) and particle approximation (by approximating q with the empirical distributions of the particles).

4. Experimental Evaluation

We compare our method with a number of baselines, including the vanilla SVGD, particle message passing (PMP), and Langevin dynamics. Note that Langevin dynamics (without the Metropolis-Hasting (MH) rejection) can also be used in decentralized settings thanks to the factorization form of the gradient $\nabla_x \log p$. However, algorithms with MH-rejection, including Metropolis-adjusted Langevin algorithm (MALA) and NUTS (Hoffman & Gelman, 2014), can not be easily performed distributedly because the MH-rejection step requires calculating a global probability ratio.

We evaluate the results on three sets of experiments, including a Gaussian MRFs toy example, a sensor localization example, and a crowdsourcing application with real-world datasets. We find our method significantly outperforms the baseline methods on *high dimensional, sparse* graphical models.

For all our experiments, we use Gaussian RBF kernel for both the vanilla and graphical SVGD and choose the bandwidth using the standard median trick. Specifically, for graphical SVGD, the kernel we use is $k_i(x, x') := \exp(-||x_{\mathcal{C}_i} - x'_{\mathcal{C}_i}||_2^2 / h_i)$ with bandwidth $h_i = med_i^2$ where med_i is the median of pairwise distances between $\{x_{\mathcal{C}_i}^\ell\}_{\ell=1}^n$ for each node x_i. We use AdaGrad (Duchi et al., 2011) for step size unless otherwise specified.

4.1. Toy example on Gaussian MRFs

We set our target distribution to be the following pairwise Gaussian MRF: $p(x) \propto \exp[\sum_{i=1}^n (b_i x_i - \frac{1}{2} A_{ii} x_i^2) - \sum_{(i,j) \in \mathcal{E}} \frac{1}{2} x_i A_{ij} x_j]$, where $\mathcal{E}$ is the edge set of a Markov graph. The model parameters (A, b) are generated first with $b_i \sim \mathcal{N}(0, 1)$, and $A_{ij} \sim \text{uniform}([-0.1, 0.1])$, followed with $A \leftarrow (A + A^\top)/2$ and $A_{ii} \leftarrow 0.1 + \sum_{j \neq i} |A_{ij}|$.

Figure 2(a)-(c) shows the results when we take $\mathcal{E}$ to be a 4-neighborhood 2D grid of size 10×10, so that the overall variable dimension d equals 100. We compare our graphical SVGD (denoted by *SVGD (graphical)*) with a number of baselines, including the typical SVGD (denoted by *SVGD (vanilla)*), exact Monte Carlo sampling, and Langevin dynamics. The results are evaluated in three different metrics:

i) Figure 2(a) shows the MSE for estimating the mean $\mathbb{E}[x_i]$ of each node i, averaged across the dimensions. We see both the graphical and vanilla SVGD perform exceptionally well in estimating the means, which does not seem to suffer from the curse of dimensionality. Graphical SVGD does not show an advantage over vanilla SVGD for the mean estimation.

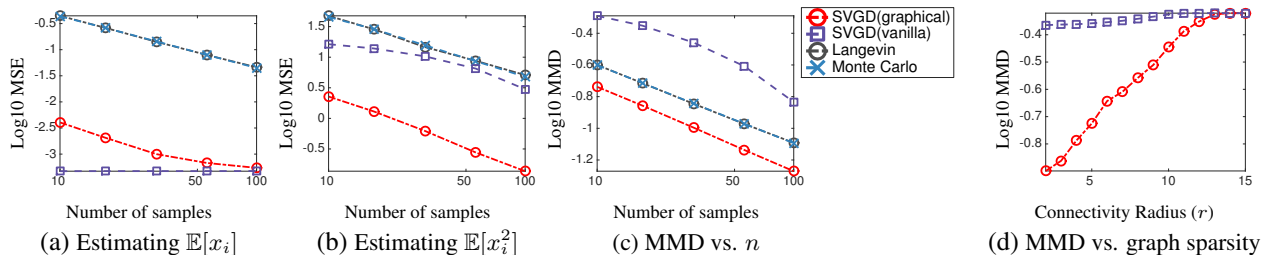ii) Figure 2(b) shows the MSE for estimating the second

Figure 2. (a)-(b): Results of Gaussian MRFs on a 10×10 4-neighborhood 2D grid, evaluated using the MSE for estimating the mean (a), the MSE for estimating the second order moments (b), and the maximum mean discrepancy between the particle and the true distribution (c). In (d), we show the performance of graphical SVGD and vanilla SVGD when the connectivity of the graph increases (with a fixed sample size of n=20). All results are averaged over 50 random trials.
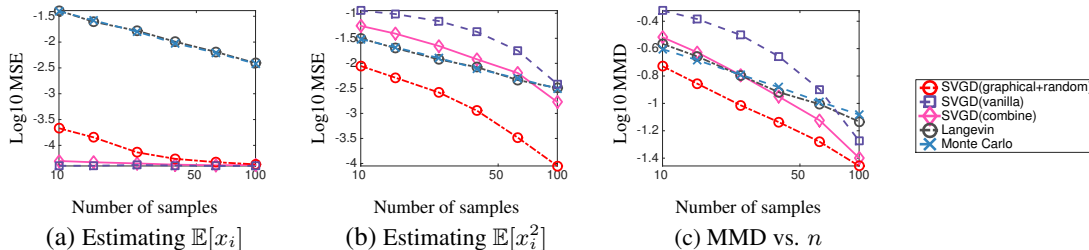

Figure 3. Results on 50 dimensional fully connected Gaussian MRFs. In this case, *SVGD (graphical+random)* uses $k_i(x, x') = k_i(x_{\mathcal{D}_i}, x'_{\mathcal{D}_i})$ where $\mathcal{D}_i$ consists of node i and four neighbors of i randomly selected at each iteration. *SVGD (combine)* uses $k_i(x, x') = (k_i(x_{\mathcal{D}_i}, x'_{\mathcal{D}_i}) + k_0(x, x'))/2$ where $k_0(x, x')$ is the regular global RBF kernel.

order moment $\mathbb{E}[x_i^2]$ of each node i, again averaged across the dimensions. In this case, graphical SVGD significantly outperforms vanilla SVGD.

iii) Figure 2(c) shows the maximum mean discrepancy (MMD) (Gretton et al., 2012) between the particles $\{x^\ell\}$ returned by different algorithms and the true distribution p, approximated by drawing a large sample from p. The kernel in MMD is taken to be the RBF kernel, with the bandwidth picked using the median trick. We find that vanilla SVGD does poorly in terms of MMD, while graphical SVGD gives the best MMD among all the methods. This is surprisingly interesting because graphical SVGD does not guarantee to match the joint distribution in theory.

Effects of Graph Sparsity Figure 2(d) shows the results of graphical and vanilla SVGD as we add more edges into the graph. The graphs are constructed by putting 10×10 points uniformly on a 2D grid, and connecting all the points with distance no larger than r, with r varying in $[1, \ldots, 14]$. The figure shows that the advantage of graphical SVGD compared to vanilla SVGD increases as the sparsity of the graph increases.

What if we apply local kernels on dense graphs? Although our method requires that the local kernels are strictly positive definite on the Markov blanket of each node, an interesting question is what would happen if the

local kernels are defined only on a subset of the Markov blanket, that is, when $k_i(x, x') = k_i(x_{\mathcal{D}_i}, x'_{\mathcal{D}_i})$ where $\mathcal{D}_i$ is a strict subset of $\mathcal{C}_i$ of p. To test this, we take p to be a fully connected Gaussian MRF with (A, b) generated in the same way as above, and test a variant of graphical SVGD, called *SVGD (graphical+random)*, which uses local kernels $k_i(x_{\mathcal{D}_i}, x'_{\mathcal{D}_i})$ where the domain $\mathcal{D}_i$ is a neighborhood of size five, consisting of node x_i and four neighbors randomly selected at each iteration. The results are shown in Figure 3, where we find that *SVGD (graphical+random)* still bring significant improvement over the vanilla SVGD in this case (see Figure 3(b)-(c)).

In Figure 3, we also tested *SVGD (combine)* whose kernel is $k_i(x, x') = \alpha k_i(x_{\mathcal{D}_i}, x'_{\mathcal{D}_i}) + (1 - \alpha)k(x, x')$, which combines the local kernel $k_i(x_{\mathcal{D}_i}, x'_{\mathcal{D}_i})$ with a global RBF kernel $k(x, x')$ (we take $\alpha = 0.5$). Compared with *SVGD (graphical+random)*, *SVGD (combine)* has the theoretical advantage that $k_i(x, x')$ is strictly integrally positive definite and hence exactly matches the joint distribution p asymptotically. Empirically, we find that the performance of *SVGD (combine)* lies in between *SVGD (graphical+random)* and *SVGD (vanilla)* as shown in Figure 3.

4.2. Sensor Network Localization

An important task in wireless sensor networks is to determine the location of each sensor given noisy measurements of pairwise distances (see, e.g., Ihler et al., 2005). We
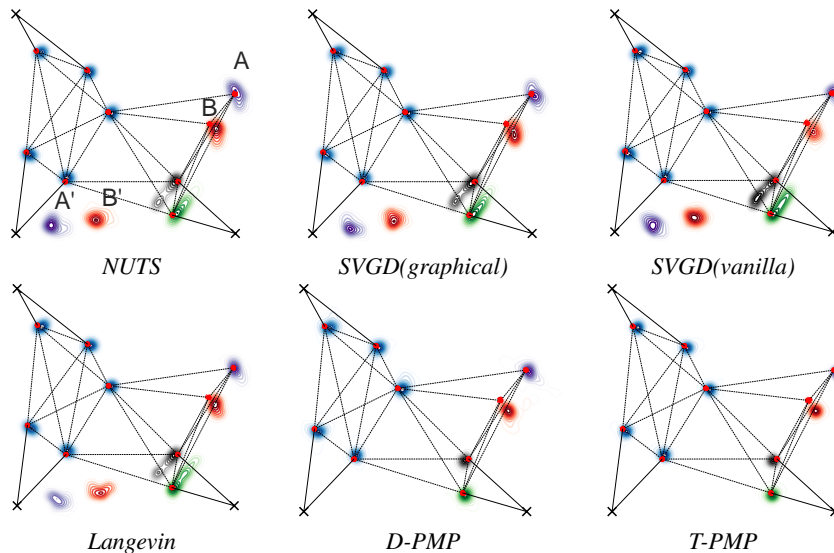
Figure 4. Results on sensor localization in a small network, with $m = 3$ anchor points (black crosses) and $d = 9$ sensors with unknown positions ($d = 9$). The contours visualize the particles given by different algorithms. Because of the lack of anchor point at the upper right corner, the posterior locations of node A and B are multi-modal, which is recovered by *NUTS*, *Langevin*, *SVGD (vanilla)*, and *SVGD(graphical)*, but not by *D-PMP* and *T-PMP*.

(a) Localization Error   (b) Estimating $\mathbb{E}[x_i]$ (c) Estimating $\mathbb{E}[x_i^2]$ (d) MMD vs. n
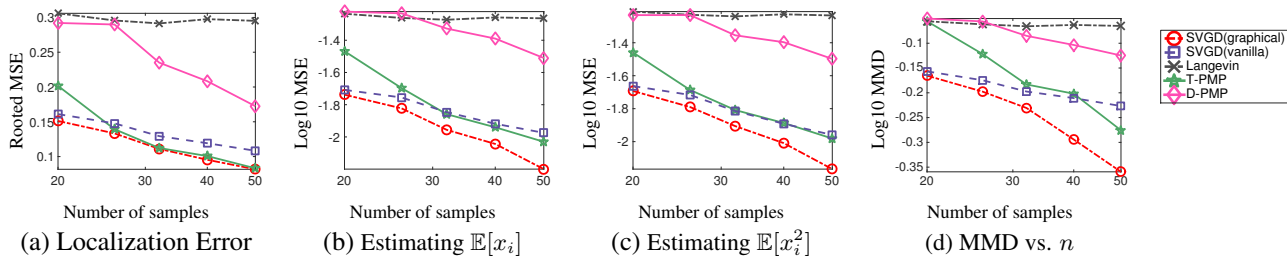
Figure 5. Results of sensor localization with $m = 4$ anchor points and $d = 100$ sensors with unknown locations. (a): the rooted mean squared error when estimating the ground truth locations using the average of the obtained particles. (b)-(d): the estimation accurate in terms of the exact posterior, evaluated against a set of sample obtained by running NUTS for a very large number of steps.

consider a 2D sensor network with nodes consisting of a set $\mathcal{S}$ of d sensors placed on unknown locations $\{x_i\}_{i \in \mathcal{S}}$, and a set $\mathcal{A}$ of m anchors with known locations $\{x_i\}_{i \in \mathcal{A}}$ where $x_i \in \mathbb{R}^2$. For our experiments, we randomly generate the true sensor locations $\{x_i^*\}_{i \in \mathcal{S}}$ uniformly from interval $[-1, 1]^2$. Assume the sensor-sensor(anchor) distances are measured with a Gaussian noise of variance σ: $r_{ij} = ||x_i - x_j||_2 + \sigma \epsilon_{ij}$ where $\epsilon_{ij} \sim \mathcal{N}(0, 1)$ and we set $\sigma = 0.05$. Assume only the pairwise distances smaller than 0.5 are measured, and denote the set of measured pairs by $\mathcal{E}$. The posterior of the unknown sensor locations is

$$p(x_{\mathcal{S}}|x_{\mathcal{A}}, r) \propto \prod_{(ij) \in \mathcal{E}} \exp \left[-\frac{1}{2\sigma^2}(||x_i - x_j||_2 - r_{ij})^2 \right].$$

Particle message passing (PMP) algorithms have been widely used for approximate inference of continuous graphical models, especially for sensor network location

(Ihler et al., 2005; Ihler & McAllester, 2009). We compare with two recent versions of PMP methods, including *T-PMP* (Besse et al., 2014) and *D-PMP* (Pacheco et al., 2014). We use the settings suggested in Pacheco & Sudderth (2015), utilizing a combination of proposals with 75% neighbor-based proposals and 25% Gaussian random walk proposals in the augmentation step. The variance of Gaussian proposals is chosen to be 0.05, which is the best setting we found on a separate validation dataset simulated with the generative model that we assumed. We also select the best learning rate for *Langevin*, *SVGD (vanilla)* and *SVGD(graphical)* in the aforementioned validation dataset.

Figure 4 reports the contours of 50 particles returned by different approaches on a small sensor network of size $d = 9, m = 3$, which includes three anchor points put on the corners. We observe that SVGD and Monte Carlo-type methods tend to capture multiple modes when the location information is ambiguous, while D-PMP and T-PMP obtain

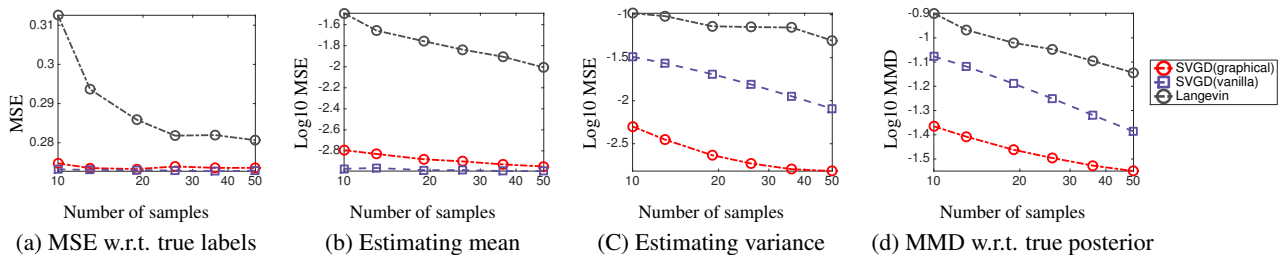| (a) MSE w.r.t. true labels | (b) Estimating mean | (C) Estimating variance | (d) MMD w.r.t. true posterior |

Figure 6. Crowdsourcing application on the *PriceUCI* dataset. (a) shows the MSE with respect to the ground truth labels, (b)-(c) the MSE in estimating the posterior mean and variance, and (d) the MMD between the particle approximation and the true posterior.

more concentrated posteriors. For example, the locations of the sensors on A and B are far away from all the three anchor points with known locations and can not be accurately estimated. As a result, the posterior includes two modes A, A' and B, B', respectively. This is correctly identified by both SVGD (graphical), SVGD (vanilla), NUTS and Langevin, but not by D-PMP and T-PMP.

We then demonstrate the effectiveness of our approach on a larger sensor network of size $d = 100$. We place four anchor sensors at the four corners $(\pm 1, \pm 1)$ $(m = 4)$, In order to evaluate the methods quantitatively, Figure 5(a) shows the mean square error between true locations and the posterior mean estimated by the average of particles. (We find the posterior is essentially unimodal in this case, so posterior mean severs a reasonable estimation). Figure 5(b)-(c) shows the approximate quantity of the posterior distribution, using a set of ground truth samples generated by NUTS (Hoffman & Gelman, 2014) with the true sensor positions as the initialization. We can see that graphical SVGD tends to outperform all the other baselines. It is interesting to see that Langevin dynamics performs significantly worse because it finds difficulty in converging well in this case (even if we searched the best step size extensively).

4.3. Application to Crowdsourcing

Crowdsourcing has been widely used in data-driven applications for collecting large amounts of labeled data. We apply our method to infer unknown continuous quantities from estimations given by the crowd workers.

Following the setting in Liu et al. (2013) and Wang et al. (2016), we assume that there is a set of questions $\{i\}$, each of which is associated with an unknown continuous quantity x_i that we want to estimate. We assume x_i is generated from a Gaussian prior $x_i \sim \mathcal{N}(0, \sigma_x^2)$. Let $\{j\}$ be a set of crowd workers that we hire to estimate $\{x_i\}$, and r_{ij} the estimate of x_i given by worker j. We assume the label r_{ij} is generated by a bias-variance Gaussian model:

$$r_{ij} = x_i + b_j + \sqrt{\nu_j}\epsilon_{ij}, \quad \epsilon_{ij} \sim \mathcal{N}(0, 1), \quad (12)$$

where b_j and ν_j are the bias and variance of worker j, respectively, both of which are unknown with a prior of $b_j \sim \mathcal{N}(0, \sigma_b^2)$ and an inverse Gamma prior $p(\nu_j) = $ Inv-Gamma(α, β) on ν_j. We are interested in evaluating the posterior estimation of $\{x_i\}$, which can be done by sampling from the joint distribution $p(x, b, \nu \mid r)$. This is a non-Gaussian, highly skewed distribution because it involves the variance parameters v_j.

We evaluate our approach on the *PriceUCI* dataset (Liu et al., 2013), which consists of 80 household items collected from the Internet, whose prices are estimated by 155 UCI undergraduate students. To construct an assignment graph for our experiments, we randomly assign 1-5 works to each question, and also ensure each worker is assigned to at least 3 questions. Because the bias b_j would not be identifiable without any ground truth, we randomly select 10 questions as control questions with known answers, and infer the labels of the remaining 70 questions. The hyper-parameters in the priors are set to be $\sigma_x = \sigma_b = 5$, $\alpha = 3, \beta = 1$. Results are averaged over 50 random trials.

We select the best learning rate for *Langenvin*, *SVGD (vanilla)* and our approach on a separate validation dataset generated with model (12). The inference is applied on model $p(\theta)$ with $\theta = [x, b, \log(\nu)]$, we clip the value of $\log(\nu)$ to $[-3, 3]$ to stablize the training. For evaluation, we generate a large set of samples for ground truth by running NUTS for a large number steps with the true task labels as initialization. As shown in figure 6, our graphical SVGD again outperforms both the typical SVGD and Langevin dynamics.

5. Conclusion

In this paper, we propose a particle-based distributed inference algorithm for approximate inference on continuous graphical models based on Stein variational gradient descent (SVGD). Our approach leverages the inherent graphical structures to improve the performance in high dimensions, and also incorporates the key advantages of gradient optimization compared to traditional PMP methods.

Acknowledgements

References

Besse, F., Rother, C., Fitzgibbon, A., and Kautz, J. PMBP: Patchmatch belief propagation for correspondence field estimation. *International Journal of Computer Vision*, 110(1):2–13, 2014.

Chwialkowski, K., Strathmann, H., and Gretton, A. A kernel test of goodness of fit. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2016.

Duchi, J., Hazan, E., and Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, 12:2121–2159, 2011.

Gorham, J. and Mackey, L. Measuring sample quality with kernels. *Proceedings of the International Conference on Machine Learning (ICML)*, 2017.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

Hoffman, M. D. and Gelman, A. The no-U-turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15(1):1593–1623, 2014.

Ihler, A. and McAllester, D. Particle belief propagation. In *Artificial Intelligence and Statistics*, pp. 256–263, 2009.

Ihler, A. T., Fisher, J. W., Moses, R. L., and Willsky, A. S. Nonparametric belief propagation for self-localization of sensor networks. *IEEE Journal on Selected Areas in Communications*, 23(4):809–819, 2005.

Lauritzen, S. L. *Graphical models*, volume 17. Clarendon Press, 1996.

Liu, Q. Stein variational gradient descent as gradient flow. In *Advances in neural information processing systems (NIPS)*, pp. 3118–3126, 2017.

Liu, Q. and Wang, D. Stein variational gradient descent: A general purpose Bayesian inference algorithm. In *Advances In Neural Information Processing Systems (NIPS)*, pp. 2370–2378, 2016.

Liu, Q., Ihler, A. T., and Steyvers, M. Scoring workers in crowdsourcing: How many control questions are enough? In *Advances in Neural Information Processing Systems (NIPS)*, pp. 1914–1922, 2013.

Liu, Q., Lee, J. D., and Jordan, M. I. A kernelized Stein discrepancy for goodness-of-fit tests. In *Proceedings of the International Conference on Machine Learning*, 2016.

Nelsen, R. B. *An introduction to copulas*. Springer Science & Business Media, 2007.

Oates, C. J., Cockayne, J., Briol, F.-X., and Girolami, M. Convergence rates for a class of estimators based on Stein's identity. *arXiv preprint arXiv:1603.03220*, 2016.

Pacheco, J. and Sudderth, E. Proteins, particles, and pseudo-max-marginals: A submodular approach. In *International Conference on Machine Learning (ICML)*, pp. 2200–2208, 2015.

Pacheco, J., Zuffi, S., Black, M., and Sudderth, E. Preserving modes and messages via diverse particle selection. In *Proceedings of the 31st International Conference on Machine Learning (ICML)*, pp. 1152–1160, 2014.

Pearl, J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers Inc., 1988.

Song, L., Gretton, A., Bickson, D., Low, Y., and Guestrin, C. Kernel belief propagation. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 707–715, 2011.

Sudderth, E. B., Ihler, A. T., Isard, M., Freeman, W. T., and Willsky, A. S. Nonparametric belief propagation. *Communications of the ACM*, 53(10):95–103, 2010.

Wainwright, M. and Jordan, M. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.*, 1(1-2):1–305, 2008.

Wang, D., Fisher III, J. W., and Liu, Q. Efficient observation selection in probabilistic graphical models using Bayesian lower bounds. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2016.

Yedidia, J. S., Freeman, W. T., and Weiss, Y. Understanding belief propagation and its generalizations. *Exploring artificial intelligence in the new millennium*, 8:236–239, 2003.

Zhuo, J., Liu, C., Shi, J., Zhu, J., Chen, N., and Zhang, B. Message passing Stein variational gradient descent. *arXiv preprint arXiv:1711.04425v2*, 2018.