Variational Message Passing Neural Network for Maximum-A-Posteriori (MAP) Inference

Abstract

Maximum-A-Posteriori (MAP) inference is a fundamental task in probabilistic inference and belief propagation (BP) is a widely used algorithm for MAP inference. Though BP has been applied successfully to many different fields, it offers no performance guarantee and often performs poorly on loopy graphs. To improve the performance on loopy graphs and to scale up to large graphs, we propose a *variational message passing neural network* (V-MPNN), where we leverage both the power of neural networks in modeling complex functions and the well-established algorithmic theories on variational belief propagation. Instead of relying on a hand-crafted variational assumption, we propose a neural free energy where a general variational distribution is parameterized through a neural network. Message passing neural networks are utilized for the minimization of neural free energy. Training of MPNNs is thus guided by neural free energy, without requiring exact MAP configurations as annotations. We empirically demonstrate the effectiveness of the proposed V-MPNN by comparing against both state-of-the-art training-free methods and training-based methods.

1 INTRODUCTION

Given a probability distribution of a set of random variables, a Maximum-A-Posteriori (MAP) inference problem involves identifying the most probable configuration of a subset of unobserved random variables with observed evidence for the rest of the variables. MAP inference problem has been studied in different communities, such as discrete energy minimization [Kappes et al., 2013] where optimization solvers are designed to directly solve for the optimal solution (i.e., the most probable configuration). Solving the MAP problem exactly is NP-hard, even with binary variables [Kolmogorov and Zabin, 2004, Cooper, 1990].

MAP inference on a probabilistic graphical model (PGM) is a fundamental task in probabilistic inference, where the joint probability distribution of a set of random variables is captured by a PGM. Different probabilistic inference algorithms have been proposed leveraging underlying structures of graphs, with belief propagation (BP) via message passing [Murphy et al., 2013] being a popular and widely used one. Besides, for efficient approximate inference, variational methods have been widely considered whereby probabilistic inference is reformulated as an optimization problem. Variational assumptions are introduced over variational distributions such as mean field assumption [Barabási et al., 1999] and Bethe assumption [Yedidia et al., 2001a]. Under mean field assumption, a variational distribution can be fully factorized which in general does not hold on an arbitrary graph. Bethe assumption is relaxed and is true on loop-free graphs. Variational BP is to perform variational inference through message passing and is theoretically grounded on the well-established connection between BP and Bethe free energy [Tatikonda and Jordan, 2002, Yedidia et al., 2003, 2000, 2001a, Heskes, 2004]. Variational BP under Bethe assumption is exact on loop-free graphs, but its performance on an arbitrary loopy graph remains inaccurate without performance guarantee [Cannings et al., 1976, Shenoy and Shafer, 2008]. Different works based on variational BP have been proposed to improve the performance on loopy graphs, all of which rely on specific variational assumptions, resulting in specific families of variational distributions.

In this work, we propose a *variational message passing neural network* (V-MPNN) for improved MAP inference performance on loopy graphs. V-MPNN leverages both the power of neural networks in modeling complex functions and the well-established algorithmic theories on variational BP. In particular, a neural free energy is proposed where variational distribution is parameterized via a neural network. An optimal variational condition is explored during training. Minimization of neural free energy is achieved through

Submitted to the 38th Conference on Uncertainty in Artificial Intelligence (UAI 2022). **To be used for reviewing only**.

message passing neural networks (MPNNs), which perform probabilistic inference through message passing. The training of V-MPNN is guided by neural free energy, different from existing neural-network-based inference methods that require exact inference results as annotations. Without requiring labeled training data, our proposed V-MPNN is data efficient. More importantly, our model can scale up to large graphs where exact inference results are unobtainable.

2 RELATED WORKS

MAP inference. MAP inference can be directly solved as an integer optimization problem [Wu et al., 2020] or can be relaxed to be a linear optimization problem (LP). With the constraints on marginals enforcing global consistency, i.e., marginal polytope, exact MAP inference can be achieved under LP relaxation [Wainwright and Jordan, 2008]. Marginal polytope is in general intractable. Instead, constraints enforcing local consistency (e.g., pairwise consistency) are considered, that is, local polytope [Sherali and Adams, 1990]. Local polytope yields pseudo-marginals that are local consistent but is not guaranteed to be exact. Unfortunately, MAP inference under LP relaxation with local polytope remains computational prohibitive, particularly on large graphs [Yanover et al., 2006].

Variational BP for MAP inference. Variational BP is to perform variational inference through message passing. Variational BP is based on the connection between BP and Bethe free energy [Yedidia et al., 2001b]. Since Bethe free energy can exactly capture only loop-free graphs, BP is guaranteed to be exact on loop-free graphs and is only an approximate inference on loopy graphs. Different techniques have been proposed to improve the performance of BP on loopy graphs, including initialization strategies [Koehler, 2019, Knoll et al., 2018], message update scheduling [Elidan et al., 2012, Knoll et al., 2015, Aksenov et al., 2020] and damping [Murphy et al., 2013, Pretti, 2005]. In addition to these practical techniques, more sophisticated hand-crafted variational distributions are proposed, leading to different variational BP algorithms [Hazan and Shashua, 2010, Riegler et al., 2012]. For example, max-product tree-reweighted message passing (TRW-MP) [Wainwright et al., 2005a] decomposed the original joint distribution into a convex combination of tree-structured distributions. A tree-reweighted variational free energy is correspondingly derived. TRW-MP is guaranteed to produce exact MAP configurations under a certain condition but it suffers from convergence issues.

Existing studies show that the entropy term within a variational free energy heavily affects the algorithm performance [Ravikumar et al., 2010, Meshi et al., 2012, Lee et al., 2020, Savchynskyy et al., 2011, Hazan and Shashua, 2012]. More specifically, when the entropy is concave and the variational free energy is thus convex, a class of message passing algorithms is obtained with convergence guarantee [Savchynskyy et al., 2011, 2012, Hazan and Shashua, 2012, Weiss et al., 2012, Meshi et al., 2015]. MAP inference error bound with convex free energy can also be derived. In this work, we propose to further reduce the MAP inference error bound by leveraging neural networks.

Neural networks for probabilistic inference. Neural networks have been considered for probabilistic inference tasks. Yoon et al. [2019] empirically demonstrated the usage of MPNN [Gilmer et al., 2017] for probabilistic inference, including MAP inference and marginal inference. The architecture of MPNNs follows a message passing scheme. Messages and beliefs are parameterized by neural networks and are learned from observed probabilistic graphs annotated with corresponding exact inference results. Though inspired by belief propagation, MPNN is solely learned from data. Different works have been proposed along this line, the majority of which are for marginal inference. Satorras and Welling [2020] proposed to refine messages from belief propagation via messages learned in MPNN. Kuck et al. [2020] proposed a belief propagation neural network (BPNN) where beliefs are regularized by minimizing a Bethe free energy. Zhang et al. [2019] proposed a factor graph neural network (FGNN) that can perform MAP inference. FGNN is proved to be equivalent to BP and thus can perform well only when ordinary BP does well. Hence, FGNN does not explicitly address the poor inference performance issue of BP on loopy graphs. All the neural-network-based methods mentioned above require either exact MAP configurations or exact partition functions as annotations for fully supervised training. As a result, these methods are limited to small graphs where exact inference results are obtainable.

3 PROPOSED METHOD

We propose a variational message passing neural network (V-MPNN) for improving inference performance on loopy graphs and scaling up to large graphs. V-MPNN leverages both the power of neural networks in modeling complex functions and the algorithmic theories on variational BP. We begin with preliminaries that are necessary for later discussions. We then introduce our proposed V-MPNN. Towards the end of this section, we summarize the training objectives of the proposed V-MPNN.

3.1 PRELIMINARIES

In this work, we focus on MAP inference on discrete pairwise markov random fields (MRFs). We first define MAP inference on MRFs and then introduce the variational free energy. We discuss different families of variational distributions and introduce the minimization of a variational free energy through message passing. In the end, we show the connection between the optimality of minimizing a varia-

tional free energy and the exactness of MAP inference.

3.1.1 MAP Inference on Markov Random Field

Given a set of N random variables $\boldsymbol{x} = \{x_1, x_2, ..., x_N\}$ in discrete space $\chi = \chi_1 \times \chi_2 \times ... \times \chi_M$, its joint probability distribution is captured by an MRF $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $|\chi_i| = k$ is the number of possible states of each variable x_i, $|\mathcal{V}| = N$, $|\mathcal{E}| = M$ with M being the total number of edges in the graph. The joint probability distribution of $\boldsymbol{x}$ is defined as,

$$p(\boldsymbol{x}) \propto \exp(\sum_{i \in \mathcal{V}} \theta_i(x_i) + \sum_{(i,j) \in \mathcal{E}} \theta_{ij}(x_i, x_j)) \quad (1)$$

$\boldsymbol{\theta}$ defines probability parameters of the graph $\mathcal{G}$. $\theta_i(x_i)$ is the unary potential of variable x_i and $\theta_{ij}(x_i, x_j)$ is the pairwise potential of two neighboring variables x_i and x_j connected via edge (i, j). Given a graph $\mathcal{G}$ and its probability parameters $\boldsymbol{\theta}$, the MAP inference task is formulated as

$$\begin{aligned} \boldsymbol{x}^* &= \arg\max_{\boldsymbol{x} \in \chi} p(\boldsymbol{x}) \\ &= \arg\max_{\boldsymbol{x} \in \chi} \sum_{i \in \mathcal{V}} \theta_i(x_i) + \sum_{(i,j) \in \mathcal{E}} \theta_{ij}(x_i, x_j) \end{aligned} \quad (2)$$

3.1.2 Variational Free Energy

Variational method converts a probabilistic inference problem to an optimziation problem, solving for a variational distribution by minimizing a variational free energy [Blei et al., 2017]. Given a target joint distribution $p(\boldsymbol{x})$, Gibbs free energy as a function of a variational distribution $q(\boldsymbol{x})$ is defined as

$$G(q) = U(q) - \mathcal{T}^\circ H(q) \quad (3)$$

$U(q) = \sum_{\boldsymbol{x}} q(\boldsymbol{x}) E(\boldsymbol{x})$ is the averaged energy. $H(q) = -\sum_{\boldsymbol{x}} q(\boldsymbol{x}) \ln q(\boldsymbol{x})$ is the entropy. $\mathcal{T}^\circ$ is the temperature. For MAP inference, temperature is specified to be a sufficiently small value ϵ ($\mathcal{T}^\circ = \epsilon$). An optimal variational distribution is obtained as

$$q^* = \arg\min_{q \in \mathbb{M}(\mathcal{G})} G(q) \quad (4)$$

Marginal polytope $\mathbb{M}(\mathcal{G})$ enforces global consistency as $\mathbb{M}(\mathcal{G}) = \{q : q \geq 0; \sum_{\boldsymbol{x}} q(\boldsymbol{x}) = 1\}$. This constrained optimization is strictly convex and q^* achieves zero KL divergence w.r.t. the target distribution, that is, $KL(q^*||p) = 0$. Exact inference can be performed with q^*. However, minimizing the Gibbs free energy over marginal polytope is in general computational prohibitive. Variational assumption is introduced for tractable variational distribution.

On pairwise MRF with the joint distribution defined in Eq. 1, we have $E(\boldsymbol{x}) = -\sum_{i \in \mathcal{V}} \theta_i(x_i) - \sum_{(i,j) \in \mathcal{E}} \theta_{ij}(x_i, x_j)$ and the averaged energy is computed as

$$\begin{aligned} &U(q) = U(\{q_i\}, \{q_{ij}\}) = \\ &-\sum_{i \in \mathcal{V}} \sum_{x_i} q_i(x_i) \theta_i(x_i) - \sum_{(i,j) \in \mathcal{E}} \sum_{x_i, x_j} q_{ij}(x_i, x_j) \theta_{ij}(x_i, x_j) \end{aligned} \quad (5)$$

The averaged energy becomes a function of local marginals $\{q_i\}_{i \in \mathcal{V}}$ and $\{q_{ij}\}_{(i,j) \in \mathcal{E}}$ with $q_i(x_i) = \sum_{\boldsymbol{x} \setminus x_i} q(\boldsymbol{x})$ and $q_{ij}(x_i, x_j) = \sum_{\boldsymbol{x} \setminus (x_i \cup x_j)} q(\boldsymbol{x})$. We thus assume a variational distribution $q(\boldsymbol{x})$ is a function of $\{q_i(x_i)\}_{i \in \mathcal{V}}$ and $\{q_{ij}(x_i, x_j)\}_{(i,j) \in \mathcal{E}}$, referred to as *pairwise assumption*. Pairwise assumption is widely used on pairwise MRF and there exist various families of variational distributions under pairwise assumption as introduced below.

Families of variational distributions. Belief propagation (BP) [Murphy et al., 2013] and TRW-MP [Wainwright et al., 2005a] are the two most representative families of variational distributions under pairwise assumption. In BP, the family of variational distribution is defined as:

$$q^{\text{BP}}(\boldsymbol{x}) = \prod_{i \in \mathcal{V}} q_i(x_i) \prod_{(i,j) \in \mathcal{E}} \frac{q_{ij}(x_i, x_j)}{q_i(x_i) q_j(x_j)} \quad (6)$$

Correspondingly, we obtain a variational free energy (i.e, Bethe free energy):

$$\begin{aligned} &G_{\text{BP}}(\{q_i\}, \{q_{ij}\}) = \\ &U(\{q_i\}, \{q_{ij}\}) - \epsilon(\sum_{i \in \mathcal{V}} (1 - |\mathcal{N}(i)|) H(q_i) + \sum_{(i,j) \in \mathcal{E}} H(q_i, q_j)) \end{aligned} \quad (7)$$

$\mathcal{N}(i)$ denotes the set of neighboring nodes of i-th node. $H(q_i) = -\sum_{x_i} q_i(x_i) \ln q_i(x_i)$. $H(q_i, q_j) = -\sum_{x_i, x_j} q_{ij}(x_i, x_j) \ln q_{ij}(x_i, x_j)$. In TRW-MP, a convex combination of tree-structured distributions via spanning trees is employed for approximating probability distribution. The family of variational distribution is defined as

$$q^{\text{TRW-MP}}(\boldsymbol{x}) = \prod_{i \in \mathcal{V}} q_i(x_i) \prod_{(i,j) \in \mathcal{E}} \left(\frac{q_{ij}(x_i, x_j)}{q_i(x_i) q_j(x_j)}\right)^{\rho_{ij}} \quad (8)$$

which is closely related to BP but differs in terms of an edge appearance probability $\rho_{ij} \in (0, 1]$. Edge appearance probability ρ_{ij} measures the probability of an edge (i, j) in a graph $\mathcal{G}$ being present in a randomly chosen spanning tree. A variational free energy is correspondingly obtained as

$$\begin{aligned} &G_{\text{TRW-MP}}(\{q_i\}, \{q_{ij}\}) = \\ &U(\{q_i\}, \{q_{ij}\}) - \epsilon(\sum_{i \in \mathcal{V}} (1 - \sum_{j \in \mathcal{N}(i)} \rho_{ij}) H(q_i) + \sum_{(i,j) \in \mathcal{E}} \rho_{ij} H(q_i, q_j)) \end{aligned} \quad (9)$$

TRW-MP is guaranteed to perform exact MAP inference under a certain post-checking condition [Wainwright et al., 2005a,b]. In summary, under the pairwise assumption, a variational free energy is of a general form:

$$\begin{aligned} &G_{\text{pairwise}}(\{q_i\}, \{q_{ij}\}) = \\ &U(\{q_i\}, \{q_{ij}\}) - \epsilon(\sum_{i \in \mathcal{V}} c_i H(q_i) + \sum_{(i,j) \in \mathcal{E}} c_{ij} H(q_i, q_j)) \end{aligned} \quad (10)$$

Each of the variational BP algorithms (e.g., BP and TRW-MP) is specific to a family of variational distributions, leading to an entropy approximation (i.e., a set of c_i and c_{ij} in Eq. 10). The performance of a variational BP algorithm is hence limited by the corresponding variational assumption. Differently, we propose to leverage the power of a neural network to automatically explore the optimal variational distribution family under the pairwise assumption.

Minimization of a variational free energy. Given a variational free energy in Eq. 10, the optimal solution set $\{q_i^*, q_{ij}^*\}_{i\in\mathcal{V},(i,j)\in\mathcal{E}}$ is obtained as:

$$\{q_i^*, q_{ij}^*\} = \arg \min_{\{q_i, q_{ij}\}\in\mathbb{L}(\mathcal{G})} G_{\text{pairwise}}(\{q_i\}, \{q_{ij}\}) \quad (11)$$

with the local polytope constraint set $\mathbb{L}(\mathcal{G}) = \{\{q_i, q_{ij}\} : q_i \geq 0; q_{ij} \geq 0; \sum_{x_i} q_i(x_i) = 1,\ \forall i \in \mathcal{V};\ q_i(x_i) = \sum_{x_j} q_{ij}(x_i, x_j),\ \forall (i,j) \in \mathcal{E}\}$. This constrained optimization is in general not convex. Its convexity depends on the concavity of the entropy term, which varies with different variational distribution families. Solving for optimal solution can be implemented through message passing. After convergence, fixed-point solutions are guaranteed to be local optimal in minimizing G_{pairwise}. However, a variational gap usually exists between q^* and the target distribution p (i.e., $KL(q^*||p) > 0$), where q^* is computed from $\{q_i^*, q_{ij}^*\}_{i\in\mathcal{V},(i,j)\in\mathcal{E}}$. MAP inference is performed as $x_i^* = \arg\max_{x_i} q_i^*(x_i)$. MAP inference is exact if there does not exist a variational gap. Otherwise, the inference remains approximate and is prone to errors.

3.2 VARIATIONAL MESSAGE PASSING NEURAL NETWORK

We now introduce the proposed *variational message passing neural network* (V-MPNN). We first introduce the proposed convex neural free energy whereby we parameterize variational distribution families via a neural network. The proposed neural free energy is provable convex. The minimal MAP inference error with the proposed neural free energy is upper bounded by an optimal entropy approximation. We then introduce the minimization of the proposed convex neural free energy through message passing neural networks (MPNNs). MPNNs perform inference through message passing with beliefs and messages parameterized via neural network parameters. In the end, we summarize the training objectives together with training procedures. The overview of V-MPNN is shown in Figure 1.

3.2.1 Convex Neural Free Energy

Under the pairwise assumption, we introduce the proposed neural free energy G_{neural}, where we parameterize variational distribution families through neural network parameters Φ. Such parameterization is implicitly achieved via a neural-network-parameterized entropy approximation:

$$\begin{aligned} &G_{\text{neural}}(\boldsymbol{q}^{node}, \boldsymbol{q}^{edge}; \Phi) \\ &= U(\boldsymbol{q}^{node}, \boldsymbol{q}^{edge}) - \epsilon H(\boldsymbol{q}^{node}, \boldsymbol{q}^{edge}; \Phi) \end{aligned} \quad (12)$$

with input tensors $\boldsymbol{q}^{node} = \{q_i\}_{i\in\mathcal{V}} \in \mathbb{R}^{N\times k}$ and $\boldsymbol{q}^{edge} = \{q_{ij}\}_{(i,j)\in\mathcal{E}} \in \mathbb{R}^{M\times k^2}$. The calculation of $U(\boldsymbol{q}^{node}, \boldsymbol{q}^{edge})$ directly follows the definition of the averaged energy and requires no free parameters to be learned. Neural-network-parameterized entropy approximation is realized through a neural network with three sets of free parameters $\boldsymbol{\phi}^{node} \in \mathbb{R}^{1\times N}$, $\boldsymbol{\phi}^{edge} \in \mathbb{R}^{1\times M}$, $\boldsymbol{\phi}^{\Delta} \in \mathbb{R}^{N\times N}$. In particular, a row-wise entropy calculation w.r.t. each input tensor is firstly performed, producing intermediate values: $\boldsymbol{h}^{node} = \{H(q_i)\}_{i\in\mathcal{V}} \in \mathbb{R}^{N\times 1}$ and $\boldsymbol{h}^{edge} = \{H(q_i, q_j)\}_{(i,j)\in\mathcal{E}} \in \mathbb{R}^{M\times 1}$. The approximate entropy is then computed as

$$\begin{aligned} &H(\boldsymbol{q}^{node}, \boldsymbol{q}^{edge}; \Phi) = \boldsymbol{\phi}^{node}\boldsymbol{h}^{node} + \\ &\texttt{exp}(\boldsymbol{\phi}^{edge})\boldsymbol{h}^{edge} + \texttt{sum}(\texttt{ReLU}(\boldsymbol{\phi}^{\Delta}) \odot \Delta\boldsymbol{h}) \end{aligned} \quad (13)$$

where $\Delta\boldsymbol{h} \in \mathbb{R}^{N\times N}$ with $\Delta h(i,j) = H(q_i, q_j) - H(q_i)$ if $(i,j) \in \mathcal{E}$. Otherwise, $\Delta h(i,j) = 0$. $\odot$ denotes element-wise product. Neural network parameters $\Phi = \{\boldsymbol{\phi}^{node}, \boldsymbol{\phi}^{edge}, \boldsymbol{\phi}^{\Delta}\}$ are unknown and are to be learned. We theoretically prove the convexity of the proposed neural free energy and the minimal MAP inference error bound through the following propositions.

Proposition 1. *Neural free energy G_{neural} is provable convex with a strictly concave neural-network-parameterized entropy approximation $H(\boldsymbol{q}^{node}, \boldsymbol{q}^{edge}; \Phi)$.*

Proof: We prove this proposition by showing the neural-network-parameterized entropy approximation is strictly concave. We first introduce the definition of concave entropy approximation [Heskes, 2004, Weiss et al., 2012]:

Definition (Concave Entropy Approximation). An approximate entropy of Eq. 10 is strictly concave over local polytope $\mathbb{L}(\mathcal{G})$ if there exist $\hat{c}_{ij} > 0$, $\hat{\alpha}_{ij} \geq 0$ and $\hat{c}_i$ such that $c_i = \hat{c}_i + \sum_{j\in\mathcal{N}(i)} \hat{\alpha}_{ij}$ and $c_{ij} = \hat{c}_{ij} - \sum_{j\in\mathcal{N}(i)} \hat{\alpha}_{ij}$. The approximate entropy becomes

$$\begin{aligned} &H(\{q_i\}, \{q_{ij}\}) = \sum_{i\in\mathcal{V}} \hat{c}_i H(q_i) + \\ &\sum_{(i,j)\in\mathcal{E}} \hat{c}_{ij} H(q_i, q_j) + \sum_{i\in\mathcal{V}} \sum_{j\in\mathcal{N}(i)} \hat{\alpha}_{ij}(H(q_i, q_j) - H(q_i)) \end{aligned} \quad (14)$$

With any set of parameters $\hat{c}_{ij} > 0$, $\hat{\alpha}_{ij} \geq 0$ and $\hat{c}_i$, the approximate entropy of Eq. 14 is strictly concave. Tensor operation defined in neural free energy (Eq. 13) is equivalent to Eq. 14, with $\boldsymbol{\phi}^{node}$, $\boldsymbol{\phi}^{edge}$ and $\boldsymbol{\phi}^{\Delta}$ corresponding to $\{\hat{c}_i\}_{i\in\mathcal{V}}$, $\{\hat{c}_{ij}\}_{(i,j)\in\mathcal{E}}$ and $\{\hat{\alpha}_{ij}\}_{i\in\mathcal{V},j\in\mathcal{N}(i)}$, respectively. $\texttt{exp}(\cdot)$ ensures the satisfaction of the constraint $\hat{c}_{ij} > 0$. $\texttt{ReLU}(\cdot)$ ensures the satisfaction of the constraint $\hat{\alpha}_{ij} \geq 0$. By definition of concave entropy approximation, the neural-network-parameterized entropy approximation

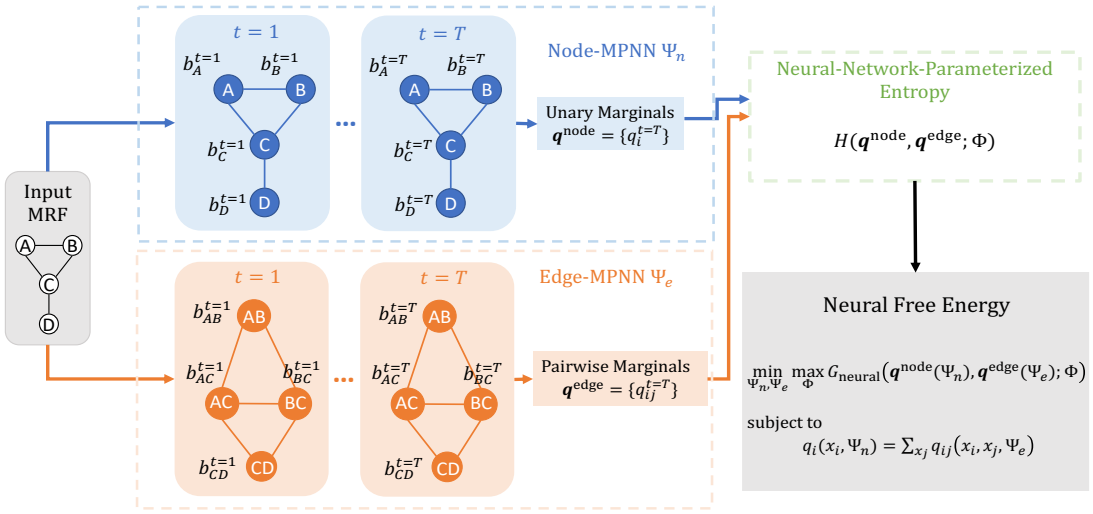

Figure 1: Overview of the proposed variational message passing neural network (V-MPNN)

$H(\boldsymbol{q}^{node}, \boldsymbol{q}^{edge}; \Phi)$ is strictly concave. The neural free energy G_{neural} is thus convex over local polytope $\mathbb{L}(\mathcal{G})$.

We now show the minimal MAP inference error with the proposed neural free energy is upper bounded by an optimal entropy approximation. We first define the MAP inference error and then present the proposition with its proof. Let $\boldsymbol{q}^*_\Phi$ denote the optimal solution set $\{q^*_{\Phi,i}, q^*_{\Phi,ij}\}_{i\in\mathcal{V},(i,j)\in\mathcal{E}}$ minimizing the neural free energy G_{neural} parameterized by Φ. Given a target probability distribution p, the MAP inference error $\Delta_{map}(\boldsymbol{q}^*_\Phi, p)$ is defined as

$$\begin{aligned}\Delta_{map}(\boldsymbol{q}^*_\Phi, p) = &\sum_{i\in\mathcal{V}}\sum_{x_i}(p_i(x_i) - q^*_{\Phi,i}(x_i))\theta_i(x_i) \\ &+ \sum_{(i,j)\in\mathcal{E}}\sum_{x_i,x_j}(p_{ij}(x_i,x_j) - q^*_{\Phi,ij}(x_i,x_j))\theta_{ij}(x_i,x_j)\end{aligned} \tag{15}$$

with $p_i = \sum_{\boldsymbol{x}\setminus x_i} p(\boldsymbol{x})$ and $p_{ij} = \sum_{\boldsymbol{x}\setminus(x_i\cup x_j)} p(\boldsymbol{x})$. By definition, $\Delta_{map}(\boldsymbol{q}^*_\Phi, p) \geq 0$ [Hazan and Shashua, 2010].

Proposition 2. *MAP inference error is upper bounded by an entropy approximation scaled by ϵ, i.e.,*

$$\Delta_{map}(\boldsymbol{q}^*_\Phi, p) \leq \epsilon H(\boldsymbol{q}^*_\Phi; \Phi) \tag{16}$$

The minimal MAP inference error is hence upper bounded by an optimal entropy approximation with $\Phi^* = \arg\min_\Phi H(\boldsymbol{q}^*_\Phi; \Phi)$.

Proof: Given the optimal solution set $\boldsymbol{q}^*_\Phi$ minimizing the neural free energy G_{neural} parameterized by Φ, we have

$$G_{\text{neural}}(\{q^*_{\Phi,i}\}, \{q^*_{\Phi,ij}\}; \Phi) \leq G_{\text{neural}}(\{p_i\}, \{p_{ij}\}; \Phi) \tag{17}$$

By reorganizing the above equation, we have

$$\Delta_{map}(\boldsymbol{q}^*_\Phi, p) \leq \epsilon(H(\boldsymbol{q}^*_\Phi; \Phi) - H(\{p_i\}, \{p_{ij}\}; \Phi)) \tag{18}$$

Given the fact that $H(\{p_i\}, \{p_{ij}\}) \geq 0$, we can have

$$\Delta_{map}(\boldsymbol{q}^*_\Phi, p) \leq \epsilon H(\boldsymbol{q}^*_\Phi; \Phi) \tag{19}$$

With an optimal set of neural network parameters $\Phi^* = \arg\min_\Phi H(\boldsymbol{q}^*_\Phi; \Phi)$, the error bound becomes

$$\Delta_{map}(\boldsymbol{q}^*_{\Phi^*}, p) \leq \epsilon H(\boldsymbol{q}^*_{\Phi^*}; \Phi^*) \tag{20}$$

We thus show that the minimal MAP inference error is upper bounded by an optimal entropy approximation. In the end, we provide a brief comparison between the proposed neural free energy and existing variational BP algorithms:

Proposition 3. *Neural free energy subsumes existing variational distribution families (e.g., BP and TRW-MP) as a strict generalization. The optimal MAP inference performance achieved with neural free energy is superior or comparable to existing variational distribution families, i.e.,* $\Delta_{map}(\boldsymbol{q}^*_{\Phi^*}, p) \leq \Delta_{map}(\boldsymbol{q}^*_{\Phi^{fix}}, p)$

Proof: By manipulating neural network parameters, different existing variational distribution families can be realized with neural free energy. For example, neural free energy with Φ specified as $\phi^{node} = 1 - |\mathcal{N}(i)|$, $\phi^{edge} = 1$ and $\phi^{\Delta} = 0$ is equivalent to BP. Furthermore, given the fact that $\boldsymbol{q}^*_{\Phi^*} = \arg\min_{\boldsymbol{q}} G_{\text{neural}}(\boldsymbol{q}; \Phi^*)$, we have

$$U(\boldsymbol{q}^*_{\Phi^*}) - \epsilon H(\boldsymbol{q}^*_{\Phi^*}; \Phi^*) \leq U(\boldsymbol{q}^*_{\Phi^{fix}}) - \epsilon H(\boldsymbol{q}^*_{\Phi^{fix}}; \Phi^*) \tag{21}$$

with $\boldsymbol{q}^*_{\Phi^{fix}}$ denotes the optimal variational distribution minimizing neural free energy specified with fixed parameters Φ^{fix}. By subtracting $U(\{p_i\}, \{p_{ij}\})$ on both sides of Eq. 21 and a re-organization, we have

$$\Delta_{map}(\boldsymbol{q}^*_{\Phi^*}, p) \leq \Delta_{map}(\boldsymbol{q}^*_{\Phi^{fix}}, p) + \epsilon\Delta \tag{22}$$

with $\Delta = H(\boldsymbol{q}^*_{\Phi^*}; \Phi^*) - H(\boldsymbol{q}^*_{\Phi^{fix}}; \Phi^*)$. If $\Delta \leq 0$, it is clear that $\Delta_{map}(\boldsymbol{q}^*_{\Phi^*}, p) \leq \Delta_{map}(\boldsymbol{q}^*_{\Phi^{fix}}, p)$. If $\Delta > 0$, we can

have $\Delta_{map}(\boldsymbol{q}^*_{\Phi^*}, p) \leq \Delta_{map}(\boldsymbol{q}^*_{\Phi^{fix}}, p)$ with a sufficiently small coefficient ($\epsilon \to 0$). Theoretically, we show that the optimal MAP inference performance achieved with neural free energy is superior or comparable to existing variational distribution families.

3.2.2 Minimization of Neural Free Energy with MPNNs

To minimize the neural free energy, we employ MPNNs. In particular, $\boldsymbol{q}^{node} = \{q_i\}_{i\in\mathcal{V}}$ and $\boldsymbol{q}^{edge} = \{q_{ij}\}_{(i,j)\in\mathcal{E}}$ are parameterized by two respective deep models: node-MPNN and edge-MPNN, leading to $\boldsymbol{q}^{node}(\Psi_n)$ and $\boldsymbol{q}^{edge}(\Psi_e)$, respectively. Both node-MPNN and edge-MPNN are constructed from MPNNs. The difference lies in the mapping from MRF to MPNN: each node in node-MPNN maps to a variable in MRF, while each node in edge-MPNN maps to an edge in MRF. We detail two modules in the following.

Node-MPNN. We map each node in node-MPNN to a variable in MRF with hidden feature $\boldsymbol{h}_i \in R^k$. k is the number of possible states of variable x_i. In total, we have node features $\boldsymbol{h} = \{\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_N\}$ and N is the total number of nodes. At every iteration t, each node receives a message from each of its neighboring nodes v_j as

$$\boldsymbol{m}^{t+1}_{j\to i} = \mathcal{M}(\boldsymbol{h}^t_i, \boldsymbol{h}^t_j) \tag{23}$$

$\mathcal{M}$ is a message function realized via a multi-layer perceptron (MLP). The messages are then aggregated through summation, i.e., $\boldsymbol{m}^{t+1}_i = \sum_{j\in\mathcal{N}(i)} \boldsymbol{m}^{t+1}_{j\to i}$. Each node then updates its hidden state with the aggregated message:

$$\boldsymbol{h}^{t+1}_i = \mathcal{U}(\boldsymbol{h}^t_i, \boldsymbol{m}^{t+1}_i) \tag{24}$$

$\mathcal{U}$ is a node update function realized through a gated recurrent unit (GRU). The update process is repeated until convergence. Estimated marginal probability of variable x_i (i.e., q_i) is obtained as

$$q_i = \sigma(\mathcal{R}(\boldsymbol{h}^{(T)}_i)) \tag{25}$$

where $\boldsymbol{h}^{(T)}_i$ is the hidden feature from the last iteration. $\mathcal{R}$ is a readout function realized through a MLP and $\sigma(x)$ refers to a softmax function. Free parameters Ψ_n of node-MPNN include parameters within its message function $\mathcal{M}$, node update function $\mathcal{U}$, and readout function $\mathcal{R}$.

Edge-MPNN. In edge-MPNN, each node is mapped to an edge in MRF with hidden feature $\boldsymbol{h}'_i \in R^{k^2}$. In total, we have node features $\boldsymbol{h}' = \{\boldsymbol{h}'_1, \boldsymbol{h}'_2, ..., \boldsymbol{h}'_M\}$ and M is the total number of edges in MRF. The same iterative update process is employed as used in node-MPNN. After the convergence, estimated pairwise marginal q_{ij} is obtained given hidden feature from the last iteration. Free parameters Ψ_e of edge-MPNN include parameters within its message function $\mathcal{M}$, node update function $\mathcal{U}$, and readout function $\mathcal{R}$.

3.2.3 Training Objectives

In summary, we have two sets of parameters to be learned: Φ and $\Psi = \{\Psi_n, \Psi_e\}$. The total training objective is based on neural free energy, i.e.,

$$\min_{\Psi}\max_{\Phi} G_{\text{neural}}(\boldsymbol{q}^{node}(\Psi_n), \boldsymbol{q}^{edge}(\Psi_e); \Phi) \tag{26}$$

under the local polytope constraint $\mathbb{L}(\mathcal{G})$. To effectively perform the training with the neural free energy, we consider a two-phase alternative update. For each iteration r, we first update $\Psi = \{\Psi_n, \Psi_e\}$ given the neural free energy specified with current Φ^r, i.e.,

$$\begin{aligned} &\Psi^{r+1}_n, \Psi^{r+1}_e = \arg\min_{\Psi_n,\Psi_e} G_{\text{neural}}(\boldsymbol{q}^{node}(\Psi_n), \boldsymbol{q}^{edge}(\Psi_e); \Phi) \\ &\text{s.t.} \sum_{x_j} q_{ij}(x_i, x_j; \Psi_e) = q_i(x_i; \Psi_n) \quad (i,j) \in \mathcal{E} \end{aligned} \tag{27}$$

Since the rest of the constraints within local polytope $\mathbb{L}(\mathcal{G})$ are naturally satisfied through a softmax function applied to the output of MPNNs, only the pairwise consistency constraint is required as suggested in the above equation. We then update Φ. By definition of G_{neural} in Eq. 12, we have $\max_\Phi G_{\text{neural}}(\Phi) = \min_\Phi H(\Phi)$. Following Proposition 2, the Φ is updated toward the direction of minimizing its corresponding MAP inference error bound, i.e.,

$$\Phi^{r+1} = \arg\min_{\Phi} H(\boldsymbol{q}^{node}(\Psi^{r+1}_n), \boldsymbol{q}^{edge}(\Psi^{r+1}_e); \Phi) \tag{28}$$

We update two sets of parameters alternatively until convergence. After training, only node-MPNN is required for MAP inference. MAP configuration is obtained via $x^*_i = \arg\max_{x_i\in\chi_i} q_i(x_i; \Psi^*_n)$.

4 EXPERIMENTS

Datasets. We consider 13 classic graphs for evaluation – these are the most representative graphs of real world models, and are employed widely in related works [Yoon et al., 2019]. Their structures are illustrated in Figure 2. There are three loop-free graphs, i.e., STAR, TREE and PATH. The other 10 graphs are loopy graphs, with the COMPLETE graph being the most complex one. To simulate graphical models with different parameters, we randomly sample from uniform distributions [Wainwright et al., 2005a]. Particularly, we assume $\theta_i(x_i) = b_i x_i$ and $\theta_{ij}(x_i, x_j) = J_{ij} x_i x_j$ with $x_i = \{-1, 1\}$. Pairwise parameters J_{ij} are sampled from a uniform distribution, i.e., $J_{ij} = J_{ji} \sim U[-1, 1]$. Unary parameters b_i are sampled from a uniform distribution as $b_i \sim U[-0.05, 0.05]$. For each type of graph, we simulate 1000 graphs for training and 100 graphs for testing. GT MAP configuration of each simulated graph is computed by enumeration. Since enumeration is a computationally expensive process, we limit the sizes of the graphs. Particularly, we consider two graph sizes: N=9 and N=15.

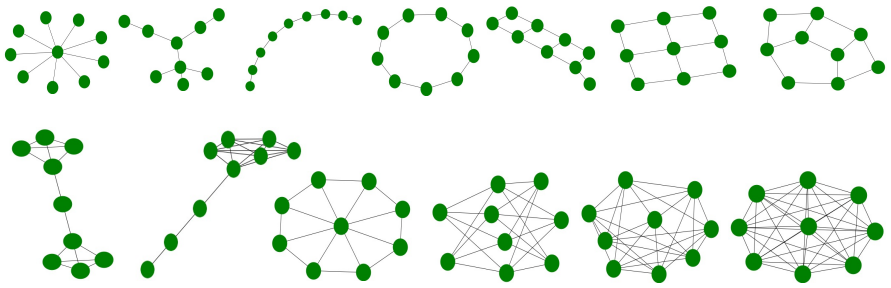

Figure 2: Structures of 13 classic graphs with 9 nodes. Graphs on the first row from left to right are: STAR, TREE, PATH, CIRCLE, LADDER, 2D GRID, CIRCULAR LADDER; graphs on the second row from left to right are: BARBELL, LOLLIPOP, WHEEL, BIPARTITE, TRIPARTITE, COMPLETE

Evaluation metrics. We employ the accuracy of estimated MAP configuration as the evaluation metric [Yoon et al., 2019]. Given a GT MAP configuration $\boldsymbol{x}^* = \{x_1^*, ..., x_N^*\}$, and an estimated MAP configuration $\hat{\boldsymbol{x}} = \{\hat{x}_1, ..., \hat{x}_N\}$, the accuracy of $\hat{\boldsymbol{x}}$ is calculated as $\frac{\#(x_i^*=\hat{x}_i)}{N}$. We report the averaged accuracy over testing graphs.

Experiment settings. ADAM optimizer is employed for training with a learning rate $1e-3$. In Eq 12, $\epsilon = 0.0001$. In Eq. 27, pairwise consistency constraints are absorbed into the training objective through Lagrangian multipliers with values 0.1. Hidden features in MPNNs are of dimension 64. Messages propagate for $T = 10$ iterations. Both node-MPNN and edge-MPNN are pre-trained with exact marginals $\{p_i\}_{i\in\mathcal{V}}$ and $\{p_{ij}\}_{(i,j)\in\mathcal{E}}$ that are obtained through enumeration.

4.1 COMPARISON TO STATE-OF-THE-ART METHODS

We compare the proposed V-MPNN to different state-of-the-art methods for approximate MAP inference. Specifically, we consider both training-free methods and training-based methods for comparison. Training-free methods refer to optimization algorithms that do not contain neural network components and thus require no training procedure, such as the belief propagation algorithm. In this work, we limit our comparisons to message-passing-based optimization approaches. Training-based methods refer to neural-network-based methods for probabilistic inference tasks.

4.1.1 Comparison to Training-free Methods

We consider three training-free methods: BP [Murphy et al., 2013], TRW-MP [Wainwright et al., 2005a] and max product linear programming (MPLP) [Globerson and Jaakkola, 2007]. For all these three methods, we apply the same stopping criterion: if the maximum number of iteration t is larger than 200 or the averaged difference between beliefs from two consecutive iterations is sufficiently small, i.e., $\frac{1}{N}\sum_{i=1}^{N}|b_i^{t+1} - b_i^t|^2 < 1e-7$, we break the algorithm and obtain the estimated inference results[1]. Following [Wainwright et al., 2005a], for both BP and TRW-MP, we apply message damping in log-space with damping parameter set to be 0.5. The edge appearance probability in TRW-MP is set as $\rho_{ij} = \frac{|\mathcal{V}|-1}{|\mathcal{E}|}$.

Results are presented in Table 1. As shown, we can see that V-MPNN achieves the best average accuracy with both sizes of graphs. On each type of graph, V-MPNN achieves overall better performance than the other three baselines. On loopy graphs, though the performance of all the algorithms decreases as the complexity of the graph increases, V-MPNN achieves better accuracy compared to the other three baselines. On COMPLETE with 15 nodes, V-MPNN achieves 78% accuracy, which is 24% higher than the accuracy achieved by TRW-BP. On loop-free graphs, such as STAR, TREE, and PATH, BP is guaranteed to produce the exact MAP configuration, and thus always achieves 100% accuracy. Though the proposed V-MPNN is theoretically shown to be a strict generalization of BP, training of MPNNs is not guaranteed to find the global optimal, leading to MAP inference errors.

4.1.2 Comparison to Training-based Methods

We compare the proposed V-MPNN to a training-based method: node-GNN [Yoon et al., 2019] for MAP inference. Node-GNN[2] is the state-of-the-art method that employs neural networks for probabilistic inference tasks. We employ the suggested hyper-parameter settings stated in the paper to perform the experiments. Results are presented in Table 2. C-LADDER denotes CIRCULAR LADDER. As shown, V-MPNN achieves significant better average accuracy with both sizes of graphs without requiring exact MAP configurations for training. On each type of graph, V-MPNN achieves overall better performance than Node-GNN. On STAR with

[1]The maximum number of iterations is set to be 200 because the number of converging runs stops changing after 200

[2]https://github.com/ks-korovina/pgm_graph_inference.

Table 1: Comparison to training-free methods

Graph	N=9				N=15			
	BP	TRW-BP	MPLP	V-MPNN	BP	TRW-BP	MPLP	V-MPNN
STAR	**1.0**	.99	**1.0**	.83	**1.0**	**1.0**	**1.0**	.84
TREE	**1.0**	.99	**1.0**	.86	**1.0**	**1.0**	**1.0**	.76
PATH	**1.0**	**1.0**	**1.0**	.86	**1.0**	**1.0**	**1.0**	.73
CYCLE	**.91**	.76	.90	.80	.84	.84	.89	.78
LADDER	.68	.66	.72	**.77**	.63	.61	.67	**.72**
2D GRID	.57	.48	.74	**.79**	.56	.50	.63	**.78**
CIRCULAR LADDER	.62	.50	.76	**.78**	.61	.53	.63	**.75**
BARBELL	.57	.55	.67	**.74**	.60	.57	.64	**.72**
LOLLIPOP	.59	.60	.61	**.81**	.62	.55	.58	**.72**
WHEEL	.56	.44	.62	**.78**	.58	.50	.62	**.69**
BIPARTITE	.54	.52	.62	**.79**	.62	.56	.55	**.75**
TRIPARTITE	.57	.62	.52	**.77**	.52	.55	.51	**.75**
COMPLETE	.56	.60	.49	**.78**	.54	.54	.53	**.78**
MEAN	.71	.67	.73	**.80**	.70	.67	.69	**.75**

Table 2: Comparison to training-based method

Graph	N=9		N=15	
	Node-GNN	V-MPNN	Node-GNN	V-MPNN
STAR	.65	**.83**	.52	**.84**
TREE	.77	**.86**	.75	**.76**
PATH	.81	**.86**	**.73**	**.73**
CYCLE	.79	**.80**	.75	**.78**
LADDER	.72	**.77**	.69	**.72**
2D GRID	.72	**.79**	.74	**.78**
C-LADDER	**.81**	.78	.71	**.75**
BARBELL	.72	**.74**	.71	**.72**
LOLLIPOP	.72	**.81**	.69	**.72**
WHEEL	.68	**.78**	**.70**	.69
BIPARTITE	.75	**.80**	.74	**.75**
TRIPARTITE	.73	**.78**	.72	**.75**
COMPLETE	**.82**	.78	.70	**.78**
MEAN	.75	**.80**	.70	**.75**

15 nodes, V-MPNN achieves 84% accuracy, which is 32% higher than the accuracy achieved by Node-GNN. These results show that, under the guidance of well-established algorithmic knowledge, the proposed V-MPNN can be trained to achieve outstanding performance, without requiring exact MAP configurations as annotations.

4.2 ABLATION STUDY

In our experiments, both node-MPNN and edge-MPNN are pre-trained with exact marginals $\{p_i\}_{i\in\mathcal{V}}$ and $\{p_{ij}\}_{(i,j)\in\mathcal{E}}$. The pre-training is mainly employed for the acceleration of training. To better analyze the influence of pre-training on MAP inference performance of V-MPNN, we perform an ablation study. Particularly, we compare the performance of V-MPNN with and without pre-training on 13 classic graphs with 9 nodes. As shown in Table 3, V-MPNN without pre-training achieves comparable average performance compared to V-MPNN with pre-training. On BARBELL, LOLLOPOP, WHEEL and TRIPARTITE, V-MPNN without pre-trianing achieves the same accuracy compared to V-MPNN with pre-trianing. From the results, we can see that pre-training introduces little effect on the inference performance of V-MPNN.

Table 3: Effectiveness of pre-training (N=9)

Graph	w/o pre-training	w pre-training
STAR	.82	**.83**
TREE	.85	**.86**
PATH	.83	**.86**
CYCLE	**.81**	.80
LADDER	**.78**	.77
2D GRID	.78	**.79**
CIRCULAR LADDER	.77	**.78**
BARBELL	**.74**	**.74**
LOLLIPOP	**.81**	**.81**
WHEEL	**.78**	**.78**
BIPARTITE	.77	**.80**
TRIPARTITE	**.78**	**.78**
COMPLETE	**.77**	.78
MEAN	.79	**.80**

5 CONCLUSION

In this work, we proposed a variational message passing neural network for MAP inference. Instead of relying on a specific family of variational distributions, we proposed a neural free energy where variational assumptions are parameterized via a neural network. An optimal family of variational distributions is learned through training. MPNNs are employed for efficient inference through message passing. Training of MPNN is performed under the guidance of neural free energy, without requiring exact MAP configurations as annotations. In our experiments, the proposed V-MPNN outperforms both state-of-the-art training-free and training-based methods for MAP inference, demonstrating the effectiveness of the proposed method.

References

Vitaly Aksenov, Dan Alistarh, and Janne H Korhonen. Relaxed scheduling for scalable belief propagation. *arXiv preprint arXiv:2002.11505*, 2020.

Albert-László Barabási, Réka Albert, and Hawoong Jeong. Mean-field theory for scale-free random networks. *Physica A: Statistical Mechanics and its Applications*, 272 (1-2):173–187, 1999.

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

C Cannings, EA Thompson, and HH Skolnick. The recursive derivation of likelihoods on complex pedigrees. *Advances in Applied Probability*, 8(4):622–625, 1976.

Gregory F Cooper. The computational complexity of probabilistic inference using bayesian belief networks. *Artificial intelligence*, 1990.

Gal Elidan, Ian McGraw, and Daphne Koller. Residual belief propagation: Informed scheduling for asynchronous message passing. *arXiv preprint arXiv:1206.6837*, 2012.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. *arXiv preprint arXiv:1704.01212*, 2017.

Amir Globerson and Tommi Jaakkola. Fixing max-product: Convergent message passing algorithms for map lp-relaxations. *Advances in neural information processing systems*, 20:553–560, 2007.

Tamir Hazan and Amnon Shashua. Norm-product belief propagation: Primal-dual message-passing for approximate inference. *IEEE Transactions on Information Theory*, 56(12):6294–6316, 2010.

Tamir Hazan and Amnon Shashua. Convergent message-passing algorithms for inference over general graphs with convex free energies. *arXiv preprint arXiv:1206.3262*, 2012.

Tom Heskes. On the uniqueness of loopy belief propagation fixed points. *Neural Computation*, 16(11):2379–2413, 2004.

Joerg Kappes, Bjoern Andres, Fred Hamprecht, Christoph Schnorr, Sebastian Nowozin, Dhruv Batra, Sungwoong Kim, Bernhard Kausler, Jan Lellmann, Nikos Komodakis, et al. A comparative study of modern inference techniques for discrete energy minimization problems. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013.

Christian Knoll, Michael Rath, Sebastian Tschiatschek, and Franz Pernkopf. Message scheduling methods for belief propagation. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 295–310. Springer, 2015.

Christian Knoll, Adrian Weller, and Franz Pernkopf. Self-guided belief propagation–a homotopy continuation method. *arXiv preprint arXiv:1812.01339*, 2018.

Frederic Koehler. Fast convergence of belief propagation to global optima: Beyond correlation decay. *Advances in Neural Information Processing Systems*, 32:8331–8341, 2019.

Vladimir Kolmogorov and Ramin Zabin. What energy functions can be minimized via graph cuts? *IEEE transactions on pattern analysis and machine intelligence*, 2004.

Jonathan Kuck, Shuvam Chakraborty, Hao Tang, Rachel Luo, Jiaming Song, Ashish Sabharwal, and Stefano Ermon. Belief propagation neural networks. *arXiv preprint arXiv:2007.00295*, 2020.

Jonathan Lee, Aldo Pacchiano, and Michael Jordan. Convergence rates of smooth message passing with rounding in entropy-regularized map inference. In *International Conference on Artificial Intelligence and Statistics*, pages 3003–3014. PMLR, 2020.

Ofer Meshi, Amir Globerson, and Tommi Jaakkola. Convergence rate analysis of map coordinate minimization algorithms. *Advances in Neural Information Processing Systems*, 25, 2012.

Ofer Meshi, Mehrdad Mahdavi, and Alex Schwing. Smooth and strong: Map inference with linear convergence. *Advances in Neural Information Processing Systems*, 28, 2015.

Kevin Murphy, Yair Weiss, and Michael I Jordan. Loopy belief propagation for approximate inference: An empirical study. *arXiv preprint arXiv:1301.6725*, 2013.

Marco Pretti. A message-passing algorithm with damping. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(11):P11008, 2005.

Pradeep Ravikumar, Alekh Agarwal, and Martin J Wainwright. Message-passing for graph-structured linear programs: Proximal methods and rounding schemes. *Journal of Machine Learning Research*, 2010.

Erwin Riegler, Gunvor Elisabeth Kirkelund, Carles Navarro Manchón, Mihai-Alin Badiu, and Bernard Henri Fleury. Merging belief propagation and the mean field approximation: A free energy approach. *IEEE Transactions on Information Theory*, 59(1):588–602, 2012.

Victor Garcia Satorras and Max Welling. Neural enhanced belief propagation on factor graphs. *arXiv preprint arXiv:2003.01998*, 2020.

Bogdan Savchynskyy, Jörg Kappes, Stefan Schmidt, and Christoph Schnörr. A study of nesterov's scheme for lagrangian decomposition and map labeling. In *CVPR 2011*. IEEE, 2011.

Bogdan Savchynskyy, Stefan Schmidt, Jörg Kappes, and Christoph Schnörr. Efficient mrf energy minimization via adaptive diminishing smoothing. *arXiv preprint arXiv:1210.4906*, 2012.

Prakash P Shenoy and Glenn Shafer. Axioms for probability and belief-function propagation. In *Classic works of the dempster-shafer theory of belief functions*, pages 499–528. Springer, 2008.

Hanif D Sherali and Warren P Adams. A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. *SIAM Journal on Discrete Mathematics*, 1990.

Sekhar C. Tatikonda and Michael I. Jordan. Loopy belief propagation and gibbs measures. In *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence*, UAI'02, 2002.

Martin J Wainwright and Michael Irwin Jordan. *Graphical models, exponential families, and variational inference*. Now Publishers Inc, 2008.

Martin J Wainwright, Tommi S Jaakkola, and Alan S Willsky. Map estimation via agreement on trees: message-passing and linear programming. *IEEE transactions on information theory*, 51(11):3697–3717, 2005a.

Martin J Wainwright, Tommi S Jaakkola, and Alan S Willsky. A new class of upper bounds on the log partition function. *IEEE Transactions on Information Theory*, 51 (7):2313–2335, 2005b.

Yair Weiss, Chen Yanover, and Talya Meltzer. Map estimation, linear programming and belief propagation with convex free energies. *arXiv preprint arXiv:1206.5286*, 2012.

Baoyuan Wu, Li Shen, Tong Zhang, and Bernard Ghanem. Map inference via l2-sphere linear program reformulation. *International Journal of Computer Vision*, 128(7):1913–1936, 2020.

Chen Yanover, Talya Meltzer, Yair Weiss, Kristin P Bennett, and Emilio Parrado-Hernández. Linear programming relaxations and belief propagation–an empirical study. *Journal of Machine Learning Research*, 7(9), 2006.

Jonathan S Yedidia, William T Freeman, Yair Weiss, et al. Generalized belief propagation. In *NIPS*, volume 13, pages 689–695, 2000.

Jonathan S Yedidia, William T Freeman, and Yair Weiss. Bethe free energy, kikuchi approximations, and belief propagation algorithms. *Advances in neural information processing systems*, 13:689, 2001a.

Jonathan S Yedidia, William T Freeman, and Yair Weiss. Generalized belief propagation. In *Advances in neural information processing systems*, pages 689–695, 2001b.

Jonathan S Yedidia, William T Freeman, Yair Weiss, et al. Understanding belief propagation and its generalizations. *Exploring artificial intelligence in the new millennium*, 8: 236–239, 2003.

KiJung Yoon, Renjie Liao, Yuwen Xiong, Lisa Zhang, Ethan Fetaya, Raquel Urtasun, Richard Zemel, and Xaq Pitkow. Inference in probabilistic graphical models by graph neural networks. In *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, pages 868–875. IEEE, 2019.

Zhen Zhang, Fan Wu, and Wee Sun Lee. Factor graph neural network. *arXiv preprint arXiv:1906.00554*, 2019.