Approximating the Sum Operation for Marginal-MAP Inference

Qiang Cheng, Feng Chen, Jianwu Dong, Wenli Xu
Tsinghua National Laboratory for Information Science and Technology
Department of Automation, Tsinghua University
Beijing 100084, China
{cheng-q09@mails, chenfeng@mail,
djw10@mails, xuwl@mail}.tsinghua.edu.cn

Alexander Ihler
Information and Computer Science
University of California, Irvine
Irvine, CA 92637-3435
ihler@ics.uci.edu

Abstract

We study the marginal-MAP problem on graphical models, and present a novel approximation method based on direct approximation of the sum operation. A primary difficulty of marginal-MAP problems lies in the non-commutativity of the sum and max operations, so that even in highly structured models, marginalization may produce a densely connected graph over the variables to be maximized, resulting in an intractable potential function with exponential size. We propose a chain decomposition approach for summing over the marginalized variables, in which we produce a structured approximation to the MAP component of the problem consisting of only pairwise potentials. We show that this approach is equivalent to the maximization of a specific variational free energy, and it provides an upper bound of the optimal probability. Finally, experimental results demonstrate that our method performs favorably compared to previous methods.

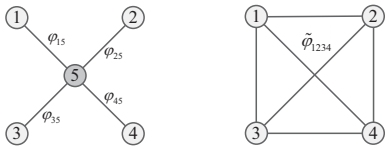
Introduction

Graphical models provide an explicit and compact representation for probability distributions that exhibit factorization structure. They are powerful tools for modeling uncertainty in the field of artificial intelligence, computer vision, bioinformatics, signal processing, and many others. Many such applications can be reduced to basic probabilistic inference tasks; typical tasks include computing marginal probabilities (sum-inference), finding the maximum a posteriori (MAP) estimate (max-inference) and marginal-MAP inference (max-sum-inference).

The marginal-MAP problem first marginalizes over a subset of the variables (sum operation), and then seeks the MAP estimate for the rest of the model variables (max operation). Marginal-MAP inference is NP^{PP}-complete, and harder than either max-inference or sum-inference (Park and Darwiche 2004). Part of the difficulty of marginal-MAP inference lies in the non-commutativity of the sum and the max operations, which can prevent "efficient" elimination orders; even for tree-structured graphical models, it can be computationally intractable (Park and Darwiche 2004; Koller and Friedman 2010).

There has been relatively little work on approximating the marginal-MAP problem until recently. State-of-the-art methods include sampling methods, search methods and message passing methods. Doucet, Godsill, and Robert (2002) propose a simple Markov chain Monte Carlo (MCMC) strategy for marginal-MAP estimates. Johansen, Doucet, and Davy (2008) sample from a sequence of artificial distributions using a sequential Monte Carlo approach. de Campos, Gámez, and Moral (1999) present a genetic algorithm to perform marginal-MAP inference. Park and Darwiche (2004) investigate belief propagation for the approximate sum-inference, and use local search for the approximate max-inference. Huang, Chavira, and Darwiche (2006) propose a branch-and-bound search method for exact marginal-MAP inference by computing the bounds on a compiled arithmetic circuit representation. Dechter and Rish (2003) propose a mini-bucket scheme for the marginal-MAP problem by partitioning the potentials into groups during elimination, and Meek and Wexler (2011) propose a related approximate variable elimination scheme that directly approximates the results of each elimination with a product of functions, bounding the error between the correct and approximate potentials. Recently, researchers have also studied marginal-MAP inference from the perspective of free energy maximization, and proposed message passing approximation algorithms. For example, Jiang, Rai, and Daumé III (2011) propose a hybrid message passing algorithm motivated by a Bethe-like free energy. Liu and Ihler (2011b) provide a general variational framework for marginal MAP, and derive several approximate inference algorithms based on the Bethe and tree-reweighted approximations; the tree-reweighted approximation provides an upper bound of the optimal energy.

In this paper, we explore a two-step approximation methods for marginal-MAP inference, in which we construct an explicit factorized approximation of the marginalized distribution using a form of approximate variable elimination, producing a structured MAP problem that can be solved using a variety of existing methods, such as dual decomposition (Sontag, Globerson, and Jaakkola 2011). We use a novel chain decomposition approach to construct the approximate marginalization, and apply a Hölder inequality (Liu and Ihler 2011a) to obtain bounds on the exact marginalization. This also allows us to interpret our method in terms of an up-



(a) (b)

Figure 1: The illustration of the marginal-MAP problem. (a) is the original graph, with a sum node (shaded) and the max nodes (unshaded). (b) is the complete graph after summing over the sum node.

per bounding variational free energy (Liu and Ihler 2011b). We show in experiments that our approach provides better bounds, and similar estimated solutions, to recently proposed message passing approximations.

Overview of Marginal-MAP Inference

In this section, we briefly review the marginal-MAP problem on graphical models. We consider only pairwise Markov random fields (MRFs) in this paper, so that a probability distribution p defined on a graph G can be defined as

$$p(\mathbf{x}) = \frac{1}{Z_\psi} \prod_{(i,j)\in\mathcal{E}} \psi_{ij}(x_i, x_j),$$

where $\mathcal{E} \subset \mathcal{V}\times\mathcal{V}$ is the set of edges, and $\mathcal{V} = \{1, 2, \ldots, N\}$ is the set of nodes. It is often useful to express $p(\mathbf{x})$ in the *overcomplete exponential family* form, by defining $\psi_{ij}(x_i, x_j) = \exp\left[\theta_{ij}(x_i, x_j)\right]$, so that

$$p(\mathbf{x}) = \exp\Big(\sum_{(i,j)\in\mathcal{E}} \theta_{ij}(x_i, x_j) - A(\boldsymbol{\theta})\Big),$$

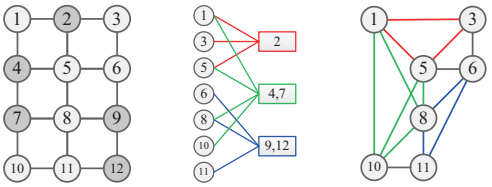
where $\boldsymbol{\theta} = \{\boldsymbol{\theta}_{ij} : (i,j) \in \mathcal{E}\}$, and $A(\boldsymbol{\theta}) = \log Z_\psi = \log\sum_{\mathbf{x}} \exp\boldsymbol{\theta}(\mathbf{x})$. As is common, we abuse notation slightly to refer to $\boldsymbol{\theta}$ and θ_{ij} as both functions of $\mathbf{x}$ and as vectors defined by the values of those functions.

Marginal-MAP inference seeks the MAP estimate for a subset of the variables ("max" variables) by marginalizing over the rest of the model variables ("sum" variables). The nodes $\mathcal{V}$ on the graphical model are thus partitioned into two sets: the sum nodes $\mathcal{V}_s$ and the max nodes $\mathcal{V}_m$, with $\mathcal{V} = \{\mathcal{V}_s, \mathcal{V}_m\}$. The edges can be divided into three types: *sum*↔*sum* (denoted $\mathcal{E}_{ss}$), *max*↔*sum* ($\mathcal{E}_{ms}$) and *max*↔*max* ($\mathcal{E}_{mm}$). The marginal-MAP problem is represented as

$$p^* = \max_{\mathbf{x}_m} \sum\nolimits_{\mathbf{x}_s} p(\mathbf{x}_s, \mathbf{x}_m), \tag{1}$$

where $\mathbf{x}_s$, $\mathbf{x}_m$ are the variables corresponding to $\mathcal{V}_s$, $\mathcal{V}_m$.

Much of the difficulty of marginal-MAP inference lies in the non-commutativity of the sum and the max operations. That is to say, we must first sum over variables $\mathbf{x}_s$, and then seek the MAP estimate for variables $\mathbf{x}_m$. For many models, such as the simple tree in Figure 1, the summation operator induces a dense, perhaps even complete graph over the max nodes, which requires exponential complexity in the number of max nodes to express.



(a) (b) (c)

Figure 2: The illustration of the bigraph view of marginal-MAP inference. (a) is the original graph, with the sum nodes (shaded) and the max nodes (unshaded). (b) is the bipartite graph of the sum nodes (right) and the max nodes (left). (c) is the graph after summing over the sum nodes.

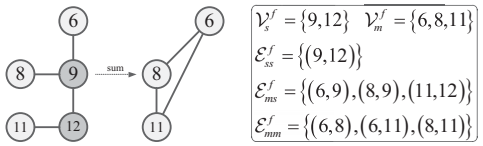


Figure 3: An example of the subgraph G^f (left).

Bigraph View of Marginal-MAP

In this section, we represent the graphical model for marginal-MAP inference by a bipartite graph, which will be helpful during the subsequent exposition.

A bipartite graph (or bigraph) is a graph whose vertices can be divided into two disjoint sets $\mathbf{U}$ and $\mathbf{V}$ such that every edge connects a vertex in $\mathbf{U}$ to one in $\mathbf{V}$ (Bondy and Murty 2008). Let the edge set $\mathcal{E}_{mf} = \mathcal{E}_{ms}$ be the edges of the bigraph, and let the node set $\mathbf{U} = \mathcal{V}_m$. We then construct "factors" f corresponding to the sum nodes, with each factor representing a set of connected nodes in $\mathbf{V}$; if there is an edge between $i \in \mathcal{V}_m$ and $j \in \mathcal{V}_s$ in graph G, then in the bigraph there is an edge between $i \in \mathbf{U}$ and the factor $f \in \mathbf{V}$ with $j \in f$. In essence, this structure represents the factor graph that would be induced by the elimination of the sum nodes in G, and we refer to these factors as *sum factors*. Figure 2 gives an illustration of the bipartite factor graph for a model with three disconnected subgraphs of sum nodes, along with the Markov random field induced by eliminating the sum nodes.

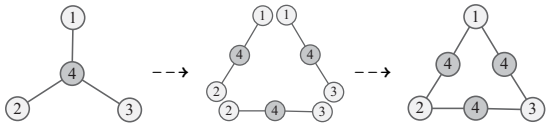
Our approximation algorithm operates on each of the sum factors independently; thus without loss of generality in the following we consider only a subgraph G^f consisting of the sum nodes $\mathcal{V}_s^f$ in a single factor f, the max nodes $\mathcal{V}_m^f$ connected to f in the bigraph, and the edges in $\mathcal{E}_{ss}^f$ and $\mathcal{E}_{ms}^f$. This means that, when the nodes $\mathcal{V}_s^f$ are eliminated, we will induce a fully connected graph over the remaining max nodes $\mathcal{V}_m^f$.

Figure 3 gives an example for a graph G^f corresponding to a subgraph of Figure 2(a) with $f = \{9, 12\}$. The potential on graph G^f is

$$\psi_f(\mathbf{x}^f) = \psi_f(\mathbf{x}_s^f, \mathbf{x}_m^f) = \prod_{(i,j)\in\mathcal{E}_{ss}^f\cup\mathcal{E}_{ms}^f} \psi_{ij}(x_i, x_j). \tag{2}$$

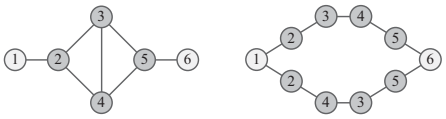
Chain Decomposition of Sum Factor

In this section, we introduce a transformation of the original model G that will be used to construct our approximate



(a) (b) (c)

Figure 4: The chain decomposition of graph G^f.



(a) (b)

Figure 5: An example that two chains are needed to cover the original graph.

marginalization, and control its computational complexity. We use the common semantics of variable splitting, or introducing copies of variables that are constrained to take on equal values, and re-parameterization, or allocating the functions defined on those copies such that the overall distribution remains invariant, to define our transformation.

Consider $\psi_f(\mathbf{x}^f)$, which is a product of the pairwise potentials on the edges of G^f. We represent $\psi_f(\mathbf{x}^f)$ as a product of chain potentials, each of which is defined on a chain between two nodes in $\mathcal{V}_m^f$. This re-representation is

$$\psi_f\left(\mathbf{x}_s^f, \mathbf{x}_m^f\right) = \prod_{(i,j)\in\mathcal{E}_{mm}^f} \tilde{\psi}_{ij}\left(\mathbf{x}_s^f, x_i, x_j\right), \tag{3}$$

where $\mathcal{E}_{mm}^f$ denotes the set of edges between two nodes $\{i, j\}$ in $\mathcal{V}_m^f$, and $\tilde{\psi}_{ij}\left(\mathbf{x}_s^f, x_i, x_j\right)$ denotes the potential defined on the chain between nodes i and j. Eq. (2) and Eq. (3) represent the same potential using different factorization forms. To achieve this transformation, we first decompose graph G^f into a set of chains, with their two ends being max nodes. These chains should be a covering (Bondy and Murty 2008) of the graph G^f. Then, we distribute the original potentials of G^f to the potentials on the chains. Finally, we combine all the max nodes with the same label into one node. Thus, we re-represents $\psi_f\left(\mathbf{x}^f\right)$ with the product of chain potentials. The following example illustrates our representation.

Example: *Consider the graph shown in Figure 4(a), where the shaded and unshaded nodes denote the sum and max nodes respectively, so that* $\mathcal{V}_s^f = \{4\}$ *and* $\mathcal{V}_m^f = \{1, 2, 3\}$. *Then,*

$$\psi_f\left(\mathbf{x}_s^f, \mathbf{x}_m^f\right) = \psi_{14}\psi_{24}\psi_{34}.$$

Let

$$\begin{aligned}
\tilde{\psi}_{12}\left(x_1, x_2, x_3, x_4\right) &= \left(\psi_{14}\right)^{\frac{1}{2}}\left(\psi_{24}\right)^{\frac{1}{2}} \\
\tilde{\psi}_{13}\left(x_1, x_2, x_3, x_4\right) &= \left(\psi_{14}\right)^{\frac{1}{2}}\left(\psi_{34}\right)^{\frac{1}{2}} \\
\tilde{\psi}_{23}\left(x_1, x_2, x_3, x_4\right) &= \left(\psi_{24}\right)^{\frac{1}{2}}\left(\psi_{34}\right)^{\frac{1}{2}},
\end{aligned}$$

and we can conclude that

$$\psi_f\left(\mathbf{x}_s^f, \mathbf{x}_m^f\right) = \tilde{\psi}_{12}\tilde{\psi}_{13}\tilde{\psi}_{23},$$

The graph representation for $\tilde{\psi}_{12}\tilde{\psi}_{13}\tilde{\psi}_{23}$ *is shown in Figure 4(c).* □

An immediate question for this representation is, how many chains are needed to cover graph G^f? Since Eq. (3) involves one term per pair of nodes in $\mathcal{V}_m^f$, it is reasonable to expect this many chains. However, this is not always the case; for some graphs fewer chains are sufficient, while for others more are required. Figure 5(b) shows an example in which a single pair of max nodes requires more than one chain to cover the graph. However, without loss of generality, we will assume one chain per pair of max nodes i, j, and associate the chain with edge (i, j) in the marginalized model.

The advantage of the chain-based representation of Eq. (3) is that each sum node copy now has at most two neighbors. By relaxing the constraints on equality among copies, we can obtain an upper bound, while each sum node copy can be eliminated efficiently.

An Upper Bound of Complete Potential

Within the subgraph G^f, the marginalization operator will produce fully connected or complete graph K_m^f over the max nodes $\mathcal{V}_m^f$, resulting in a computationally intractable function (referred to as the complete potential). In this section, we will approximate the complete potential with a product of pairwise potentials that are defined on the edges of the complete graph. Furthermore, we design this approximation so that it provides an upper bound of the complete potential.

Using the chain-structured covering designed in the previous section, each sum node copy is associated with some chain with two max node endpoints, say i and j. We assign a weight ω_{ij} to this copy, with $0 \leq \omega_{ij} \leq 1$ and $\sum_{(i,j)\in\mathcal{E}_{mm}^f} \omega_{ij} = 1$. We can then approximate the complete potential using an approximate elimination based on Hölder's inequality (see (Liu and Ihler 2011a)), so that

$$\psi_f(\mathbf{x}_m^f) = \sum_{\mathbf{x}_s^f} \psi_f(\mathbf{x}_s^f, \mathbf{x}_m^f) \approx \prod_{(i,j)\in\mathcal{E}_{mm}^f} \tilde{\psi}_{ij}(x_i, x_j) \tag{4}$$

where $\tilde{\psi}_{ij}(x_i, x_j)$ is defined as

$$\tilde{\psi}_{ij}(x_i, x_j) = \left(\sum\nolimits_{\mathbf{x}_s^f} \tilde{\psi}_{ij}(\mathbf{x}_s^f, x_i, x_j)^{\frac{1}{\omega_{ij}}}\right)^{\omega_{ij}}. \tag{5}$$

Because $\tilde{\psi}_{ij}(\mathbf{x}_s^f, x_i, x_j)$ is chain-structured, it can be computed efficiently in $\mathcal{O}\left(Nd^3\right)$, where N is the number of variables on the chain and d is the number of states for each variable. The overall complexity for computing the pairwise potentials of the complete graph K_m^f is no more than $\frac{\left|\mathcal{E}_{mm}^f\right|^2}{2}\mathcal{O}\left(\left|\mathcal{V}_s^f\right| d^3\right)$.

Eq. (4) also provides an upper bound on the true complete potential:

Theorem 1. *The product of the pair-wise potentials yields an upper bound of the true complete potential, that is*

$$\psi_f\left(\mathbf{x}_m^f\right) \leq \prod_{(i,j)\in\mathcal{E}_{mm}^f} \tilde{\psi}_{ij}\left(x_i, x_j\right), \tag{6}$$

where $\tilde{\psi}_{ij}\left(x_i, x_j\right)$ *is defined as in Eq. (5). Equality holds if* $\forall\, \mathbf{x}_s^f \in \mathcal{X}_s^f, \mathbf{x}_m^f \in \mathcal{X}_m^f$

$$\frac{\tilde{\psi}_{ij}\left(\mathbf{x}_s^f, x_i, x_j\right)^{\frac{1}{\omega_{ij}}}}{\sum\limits_{\mathbf{x}_s^f}\tilde{\psi}_{ij}\left(\mathbf{x}_s^f, x_i, x_j\right)^{\frac{1}{\omega_{ij}}}} = const., \tag{7}$$

where $(i,j) \in \mathcal{E}_{mm}^f$*, and* $const.$ *denotes the same constant.*

Proof. The result follows directly from Hölder's inequality. Given $f_i(\mathbf{x}) \geq 0, 0 \leq \omega_i \leq 1, i = \{1,2,\ldots,n\}$, and $\sum\limits_{i=1}^{n}\omega_i = 1$, Hölder's inequality (Hardy, Littlewood, and Pólya 1988) states that

$$\sum_{\mathbf{x}_a}\prod_{i=1}^{n} f_i(\mathbf{x})^{\omega_i} \leq \prod_{i=1}^{n}\left(\sum_{\mathbf{x}_a} f_i(\mathbf{x})\right)^{\omega_i}. \tag{8}$$

Taking $f_i(\mathbf{x}) = \tilde{\psi}_{ij}\left(\mathbf{x}_s^f, x_i, x_j\right)^{\frac{1}{\omega_{ij}}}$, and the definition of the complete potential in Eq. (4), the r.h.s. of Eq. (6) yields the definition Eq. (5). The condition in Eq. (7) can be derived from the equality condition of Hölder's inequality. □

In effect, this replaces the complete graph induced by eliminating connected component G^f with a pair-wise graphical model that upper bounds the original.

Dual Decomposition for MAP

By summing over all the sum variables using Eq. (5), we obtain a graph with only the max variables. The next step is to estimate the maximum a posteriori (MAP) configuration of these variables.

The MAP problem can be solved efficiently using the technique of dual decomposition (Sontag, Globerson, and Jaakkola 2011). For easy implementation, we can use tree-decomposed block coordinate descent algorithms, such as the max-sum diffusion (MSD) algorithm (Werner 2007), the max product linear programming (MPLP) algorithm (Globerson and Jaakkola 2008) or the sequential tree-reweighted message passing (TRW-S) algorithm (Kolmogorov 2006). To obtain tighter bounds, we can use algorithms with high-order constraints, such as the generalized MPLP (GMPLP) algorithm (Sontag et al. 2008) or outer-planar decompositions (Batra et al. 2010).

Under the framework of dual decomposition, the above algorithms yield an upper bound on the MAP assignment. Recall that the chain decomposition approach returns an upper bound for the complete potential; thus we conclude that the approximation approach based on chain decomposition and dual decomposition yields an upper bound of p^* in Eq. (1). We give a sketch of our algorithm for solving the marginal-MAP problem in Algorithm 1.

Variational Representation

Our algorithm can also be interpreted in a variational framework, using the connection between Hölder's inequality and weighted entropy decompositions (Liu and Ihler 2011a).

Liu and Ihler (2011b) provide a variational framework for addressing the marginal-MAP problem. Considering only the graph G^f, the variational representation $\Phi(\boldsymbol{\theta})$ on G^f is

Algorithm 1 The Chain Decomposition Algorithm

Input: A graphical model G for marginal-MAP inference.
Output: An upper bound of p^* in Eq. (1).

1: Represent the potentials of the sum nodes with the potentials on a set of chains using Eq. (3).
2: Sum over the sum variables using Eq. (5).
3: Use a dual decomposition technique for MAP inference.
4: Return the upper bound and the MAP estimate.

$$\Phi\left(\boldsymbol{\theta}^f\right) = \max_{\boldsymbol{\mu}^f \in \mathcal{M}^f}\left\{\left\langle\boldsymbol{\theta}^f, \boldsymbol{\mu}^f\right\rangle + H\left(\mathbf{x}_s^f|\mathbf{x}_m^f; \boldsymbol{\mu}^f\right)\right\}, \tag{9}$$

where $\mathcal{M}^f$ is the marginal polytope, and $H\left(\mathbf{x}_s^f|\mathbf{x}_m^f; \boldsymbol{\mu}^f\right) = -\sum_{\mathbf{x}^f} q_{\boldsymbol{\mu}}\left(\mathbf{x}^f\right)\log q_{\boldsymbol{\mu}}\left(\mathbf{x}_s^f|\mathbf{x}_m^f\right)$ is the conditional entropy, with $q_{\boldsymbol{\mu}}\left(\mathbf{x}^f\right)$ being the maximum entropy distribution corresponding to $\boldsymbol{\mu}^f$. The variational representation is an equivalent transformation of the original marginal-MAP problem, with $\Phi\left(\boldsymbol{\theta}^f\right) = \log p_f^*$, where p_f^* is defined on graph G^f as in Eq. (1). However, this dual representation does not reduce the computational cost.

For our purposes, it is more convenient to express the variational form on the sum nodes alone, keeping the optimization over $\mathbf{x}_m^f$ in its combinatorial form:

$$\Phi\left(\boldsymbol{\theta}^f\right) = \max_{\mathbf{x}_m^f \in \mathcal{X}_m^f}\left\{\left\langle\boldsymbol{\theta}\left(\mathbf{x}_m^f\right), \mathbf{x}_m^f\right\rangle + \Phi\left(\boldsymbol{\theta}\left(\mathbf{x}_s^f|\mathbf{x}_m^f\right)\right)\right\}, \tag{10}$$

where $\forall\ \mathbf{x}_m^f \in \mathcal{X}_m^f$, $\Phi\left(\boldsymbol{\theta}\left(\mathbf{x}_s^f|\mathbf{x}_m^f\right)\right)$ is defined as:

$$\Phi\left(\boldsymbol{\theta}\left(\mathbf{x}_s^f|\mathbf{x}_m^f\right)\right) = \max_{\boldsymbol{\mu}_s^f \in \mathcal{M}\left(\mathbf{x}_s^f\right)}\left\{\begin{array}{c}\left\langle\boldsymbol{\theta}\left(\mathbf{x}_s^f|\mathbf{x}_m^f\right), \boldsymbol{\mu}_s^f\right\rangle \\ +H\left(\mathbf{x}_s^f; \boldsymbol{\mu}_s^f\right)\end{array}\right\}. \tag{11}$$

The two representations in Eq. (9) and Eq. (10) are equivalent at their optimal values.

Our approximation decomposes G^f into a set of chains, with the two ends of each chain being max nodes. Let $\mathcal{C}\left(G^f\right)$ be the set of chains, and $\mathcal{C}_i^f$ be a chain in $\mathcal{C}\left(G^f\right)$. First, we decompose the parameters $\boldsymbol{\theta}\left(\mathbf{x}_s^f|\mathbf{x}_m^f\right)$ on G^f into a combination of the parameters on a set of chains, such as

$$\boldsymbol{\theta}\left(\mathbf{x}_s^f|\mathbf{x}_m^f\right) = \sum_{\mathcal{C}_i^f \in \mathcal{C}(G^f)} \omega_{\mathcal{C}_i^f}\boldsymbol{\theta}^{\mathcal{C}_i^f}\left(\mathbf{x}_s^f|\mathbf{x}_m^f\right),$$

where $\sum\omega_{\mathcal{C}_i^f} = 1$, and $\boldsymbol{\theta}^{\mathcal{C}_i^f}\left(\mathbf{x}_s^f|\mathbf{x}_m^f\right)$ is the parameter on chain $\mathcal{C}_i^f$.

Since $\Phi\left(\boldsymbol{\theta}\left(\mathbf{x}_s^f|\mathbf{x}_m^f\right)\right)$ is a convex function w.r.t. the parameter $\boldsymbol{\theta}\left(\mathbf{x}_s^f|\mathbf{x}_m^f\right)$ (Wainwright, Jaakkola, and Willsky 2005), we can apply Jensen's inequality to a convex combination of the parameter and obtain an upper bound:

$$\begin{aligned}\Phi\left(\boldsymbol{\theta}\left(\mathbf{x}_s^f|\mathbf{x}_m^f\right)\right) &= \Phi\left(\sum_{\mathcal{C}_i^f \in \mathcal{C}(G^f)} \omega_{\mathcal{C}_i^f}\boldsymbol{\theta}^{\mathcal{C}_i^f}\left(\mathbf{x}_s^f|\mathbf{x}_m^f\right)\right) \\ &\leq \sum_{\mathcal{C}_i^f \in \mathcal{C}(G^f)} \omega_{\mathcal{C}_i^f}\Phi\left(\boldsymbol{\theta}^{\mathcal{C}_i^f}\left(\mathbf{x}_s^f|\mathbf{x}_m^f\right)\right).\end{aligned}$$

Then, $\Phi\left(\boldsymbol{\theta}^f\right)$ can be approximated as

$$\tilde{\Phi}\left(\boldsymbol{\theta}^f\right) = \max_{\mathbf{x}_m^f \in \mathcal{X}_m^f} \left\{ \begin{array}{r} \left\langle \boldsymbol{\theta}\left(\mathbf{x}_m^f\right), \mathbf{x}_m^f \right\rangle + \\ \sum_{\mathcal{C}_i^f \in \mathcal{C}(G^f)} \omega_{\mathcal{C}_i^f} \Phi\left(\boldsymbol{\theta}^{\mathcal{C}_i^f}\left(\mathbf{x}_s^f | \mathbf{x}_m^f\right)\right) \end{array} \right\}. \tag{12}$$

The max operation on $\mathbf{x}_m^f$ in Eq. (12) is to solve an integer programming problem. This problem can be further approximated using the technique of linear programming relaxation or dual decomposition. Algorithm 1 provides a direct implementation of Eq. (12), then applies the dual decomposition technique for the MAP estimate component.

Relations with A-B Tree Decomposition

Based on the variational framework, Liu and Ihler (2011b) introduce a tree-reweighted free energy by decomposing the original graph into a combination of A-B trees. The A-B tree is such a tree that no two edges in $\mathcal{E}_{ms}$ are connected by nodes in $\mathcal{V}_s$. In the following, we will analyze the relations between our chain-based decomposition method and the A-B tree-based decomposition method.

Both methods use reweighted free energy approximations to provide upper bounds on the optimal marginal MAP value. However, the primary differences are:

I. If tree-decomposed block coordinate descent algorithms are used for the MAP estimate in Algorithm 1, our chain-based method provides a form of "hyper-tree" decomposition on the sum nodes, since elimination of each sum node is allowed to involve two adjacent nodes (a chain).

II. Our method does not require any particular choice of optimization for the max nodes, since an explicit pair-wise model is produced. In practice we use dual-decomposition, but other methods are easily applied.

III. Our framework explicitly selects a fixed allocation of the sum node parameters $\tilde{\psi}_{ij}(\mathbf{x}_s^f, x_i, x_j)$ to each chain, whereas the message-passing process in the A-B tree method is able to tighten its bound during the iterative process.

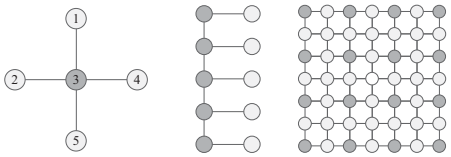
(I) suggests that, if the optimal values of the weights ω_{ij} and the re-parameterization into chains $\tilde{\psi}_{ij}(\mathbf{x}_s^f, x_i, x_j)$ are used, the chain decomposition bound will be tighter than that of a tree-reweighted collection of A-B trees.

Experiments

In this section, we conduct experiments to show the effectiveness of the chain decomposition algorithm. We test the chain decomposition algorithm on three types of graphs: star model, chain model, and grid model, as shown in Figure 6. The distribution on these models are defined as

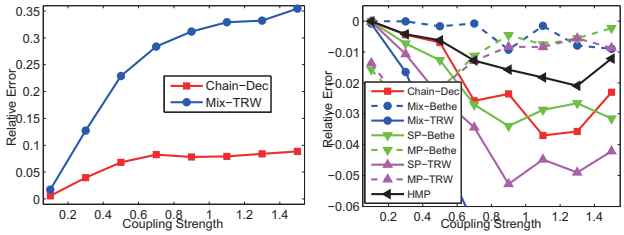
$$p(\mathbf{x}) \propto \exp\left(\sum_{i\in\mathcal{V}} \theta_i(x_i) + \sum_{(i,j)\in\mathcal{E}} \theta_{ij}(x_i, x_j)\right).$$

We set $\theta_{ij}(k,k) = 0$, and randomly generate $\theta_i(k) \sim \mathcal{N}(0, 0.1)$, $\theta_{ij}(k,l) \sim \mathcal{N}(0,\sigma)$ for $k \neq l$, where $\sigma \in \{0.1, 0.3, \ldots, 1.5\}$ is the coupling strength. For the star model and the chain model, each variable has three states,



(a) (b) (c)

Figure 6: The three models for experiments, with shaded sum nodes and unshaded max nodes. (a) star model, (b) chain model, (c) grid model.



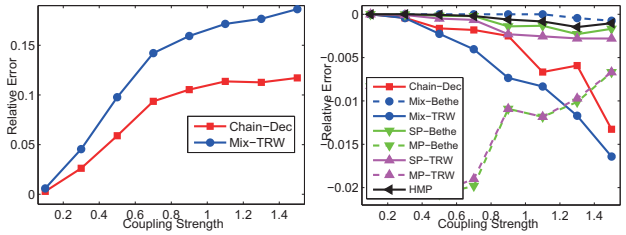
(a) (b)

Figure 7: Results on the star model of Figure 6(a). (a) The upper bounds obtained by the tree-reweighted mixed message passing and the chain decomposition algorithms; (b) The relative energy errors of different algorithms.

and for the grid model, each variable has two states. The results are obtained after averaging 100 trials.

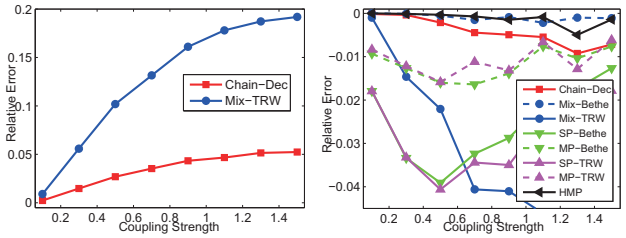
We test the Bethe mixed message passing (Mix-Bethe) (Liu and Ihler 2011b), the tree-reweighted mixed message passing (Mix-TRW) (Liu and Ihler 2011b), the hybrid message passing (HMP) (Jiang, Rai, and Daumé III 2011), the Bethe sum-product (SP-Bethe), the Bethe max-product (MP-Bethe), the tree-reweighted sum-product (SP-TRW), the tree-reweighted max-product (MP-TRW), and the chain decomposition (Chain-Dec) algorithms on these graphical models. To implement the tree-reweighted algorithms on the grid model, we decompose it into a combination of four spanning A-B trees. We compute the relative energy errors of different algorithms. The relative energy error is defined as $(\log\hat{p} - \log p^*)/\log p^*$, where $\log p^*$ is the maximal energy and $\log\hat{p}$ is the approximate energy obtained by that algorithm. Here, $\hat{p} = \sum_{\mathbf{x}_s} p(\mathbf{x}_s, \hat{\mathbf{x}}_m)$, where $\hat{\mathbf{x}}_m$ is the estimated solution by different algorithm. For the Mix-TRW and Chain-Dec algorithms, we also compute the upper bounds of the maximal energy. The results are shown in Figures 7,8,9.

Figures 7(a),8(a),9(a) show that the upper bound obtained by the chain decomposition algorithm is tighter than the upper bound obtained by the tree-reweighted mixed message passing algorithm. Figures 7(b),8(b),9(b) show that the Bethe mixed message passing algorithm and the hybrid message passing algorithm perform much better than the other algorithms. Although the chain decomposition algorithm does not give the best solution, its performance is comparable to the Bethe mixed message passing algorithm and the hybrid message passing algorithm. Moreover, the chain decomposition algorithm always gives smaller relative error than the tree-reweighted mixed message passing algorithm.



(a) (b)

Figure 8: Results on the chain model of Figure 6(b). (a) The upper bounds obtained by Mix-TRW and Chain-Dec; (b) The relative energy errors of different algorithms.



(a) (b)

Figure 9: Results on the grid model of Figure 6(c). (a) The upper bounds obtained by Mix-TRW and Chain-Dec; (b) The relative energy errors of different algorithms.

Conclusion

This paper presents a novel method to efficiently approximate the marginalization step in marginal-MAP problems on graphical models. The sum operation results in a complete potential over the connected neighborhood, with exponential size. We propose a chain decomposition approach to approximate this complete potential with a product of pair-wise potentials. This technique can be interpreted as a "reweighted" variational approach, with a corresponding free energy approximation, and returns an upper bound of the maximal energy. Experimental results show that our method gives good upper bounds when compared to existing techniques, and performs comparably to state-of-the-art methods on solution quality.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No.61071131), Beijing Natural Science Foundation (No.4122040), National Key Basic Research and Development Program of China (No.2009CB320602) and United Technologies Research Center (UTRC).

References

Batra, D.; Gallagher, A.; Parikh, D.; and Chen, T. 2010. Beyond trees: MRF inference via outer-planar decomposition. In *CVPR*.

Bondy, J., and Murty, U. 2008. *Graph Theory*. Springer Berlin.

de Campos, L.; Gámez, J.; and Moral, S. 1999. Partial abductive inference in Bayesian belief networks using a genetic algorithm. *Pattern Recogn. Lett.* 20(11-13):1211–1217.

Dechter, R., and Rish, I. 2003. Mini-buckets: A general scheme for bounded inference. *J. ACM* 50(2):107–153.

Doucet, A.; Godsill, S.; and Robert, C. 2002. Marginal maximum a posteriori estimation using Markov chain Monte Carlo. *Stat. Comput.* 12:77–84.

Globerson, A., and Jaakkola, T. 2008. Fixing max-product: Convergent message passing algorithms for MAP LP-relaxations. In *NIPS*.

Hardy, G.; Littlewood, J.; and Pólya, G. 1988. *Inequalities*. Cambridge University Press.

Huang, J.; Chavira, M.; and Darwiche, A. 2006. Solving MAP exactly by searching on complied arithmetic circuits. In *AAAI*.

Jiang, J.; Rai, P.; and Daumé III, H. 2011. Message-passing for approximate MAP inference with latent variables. In *NIPS*.

Johansen, A.; Doucet, A.; and Davy, M. 2008. Particle methods for maximum likelihood estimation in latent variable models. *Stat. Comput.* 18(1):47–57.

Koller, D., and Friedman, N. 2010. *Probablistic Graphical Models*. MIT Press.

Kolmogorov, V. 2006. Convergent tree-reweighted message passing for energy minimization. *IEEE Trans. PAMI* 28(10):1568–1583.

Liu, Q., and Ihler, A. 2011a. Bounding the partition function using Hölder's inequality. In *ICML*, 849–856.

Liu, Q., and Ihler, A. 2011b. Variational algorithms for marginal MAP. In *UAI*.

Meek, C., and Wexler, Y. 2011. Improved approximate sum-product inference using multiplicative error bounds. In *Bayesian Statistics 9*. Oxford University Press.

Park, J., and Darwiche, A. 2004. Complexity results and approximation strategies for MAP explanations. *J. Artif. Intell. Res.* 21(1):101–133.

Sontag, D.; Meltzer, T.; Globerson, A.; Jaakkola, T.; and Weiss, Y. 2008. Tightening LP relaxations for MAP using message passing. In *UAI*.

Sontag, D.; Globerson, A.; and Jaakkola, T. 2011. Introduction to dual decomposition for inference. In *Optimization for Machine Learning*. MIT Press.

Wainwright, M.; Jaakkola, T.; and Willsky, A. 2005. A new class of upper bounds on the log partition function. *IEEE Trans. Inf. Theory* 51(7):2313.

Werner, T. 2007. A linear programming approach to max-sum problem: A review. *IEEE Trans. PAMI* 1165–1179.