Exact MAP estimates by (hyper)tree agreement

Martin J. Wainwright,
Department of EECS,
UC Berkeley,
Berkeley, CA 94720
martinw@eecs.berkeley.edu

Tommi S. Jaakkola and Alan S. Willsky,
Department of EECS,
Massachusetts Institute of Technology,
Cambridge, MA, 02139
{tommi,willsky}@mit.edu

Abstract

We describe a method for computing provably exact *maximum a posteriori* (MAP) estimates for a subclass of problems on graphs with cycles. The basic idea is to represent the original problem on the graph with cycles as a convex combination of tree-structured problems. A convexity argument then guarantees that the optimal value of the original problem (i.e., the log probability of the MAP assignment) is upper bounded by the combined optimal values of the tree problems. We prove that this upper bound is met with equality if and only if the tree problems share an optimal con£guration in common. An important implication is that any such shared con£guration must also be the MAP con£guration for the original problem. Next we develop a tree-reweighted max-product algorithm for attempting to £nd convex combinations of tree-structured problems that share a common optimum. We give necessary and suf£cient conditions for a £xed point to yield the exact MAP estimate. An attractive feature of our analysis is that it generalizes naturally to convex combinations of hypertree-structured distributions.

1 Introduction

Integer programming problems arise in various £elds, including machine learning, statistical physics, communication theory, and error-correcting coding. In many cases, such problems can be formulated in terms of undirected graphical models [e.g., 1], in which the cost function corresponds to a graph-structured probability distribution, and the problem of interest is to £nd the *maximum a posteriori* (MAP) con£guration.

In previous work [2], we have shown how to use convex combinations of tree-structured distributions in order to upper bound the log partition function. In this paper, we apply similar ideas to upper bound the log probability of the MAP con£guration. As we show, this upper bound is met with equality whenever there is a con£guration that is optimal for all trees, in which case it must also be a MAP con£guration for the original problem. The work described here also makes connections with the max-product algorithm [e.g., 3, 4, 5], a well-known method for attempting to compute the MAP con£guration, one which is exact for trees but approximate for graphs with cycles. In the context of coding problems, Frey and Koetter [4] developed an attenuated version of max-product, which is guaranteed to £nd the MAP codeword if it converges. One contribution of this paper is to develop a *tree-reweighted max-product algorithm* that attempts to £nd a collection of tree-structured problems that share a common optimum. This algorithm, though similar to both the standard and attenuated max-product updates [4], differs in key ways.

The remainder of this paper is organized as follows. The next two subsections provide background on exponential families and convex combinations. In Section 2, we introduce the basic form of the upper bounds on the log probability of the MAP assignment, and then develop necessary and suf£cient conditions for it to tight (i.e., met with equality). In Section 3, we develop tree-reweighted max-product algorithms for attempting to £nd a convex combination of trees that yields a tight bound. We prove that for positive compatibility functions, the algorithm always has at least one £xed point; moreover, if a key uniqueness condition is satis£ed, the con£guration speci£ed by a £xed point must be MAP optimal. We also illustrate how the algorithm, like the standard max-product algorithm [5], can fail if the uniqueness condition is not satis£ed. We conclude in Section 4 with pointers to related work, and extensions of the current work.

1.1 Notation and set-up

Consider an undirected (simple) graph $G = (V, E)$. For each vertex $s \in V$, let x_s be a random variable taking values in the discrete space $\mathcal{X}_s = \{0, 1, \ldots, m_s - 1\}$. We use the letters j, k to denote particular elements of the sample space $\mathcal{X}_s$. The overall random vector $\mathbf{x} = \{x_s \mid s \in V\}$ takes values in the Cartesian product space $\mathcal{X}^N = \mathcal{X}_1 \times \cdots \times \mathcal{X}_N$, where $N = |V|$. We make use of the following exponential representation of a graph-structured distribution $p(\mathbf{x})$. For some index set $\mathcal{I}$, we let $\boldsymbol{\phi} = \{\phi_\alpha \mid \alpha \in \mathcal{I}\}$ denote a collection of potential functions de£ned on the cliques of G, and let $\theta = \{\theta_\alpha \mid \alpha \in \mathcal{I}\}$ be a vector of real-valued weights on these potential functions. The exponential family determined by $\boldsymbol{\phi}$ is the collection of distributions $p(\mathbf{x}; \theta) \propto \exp\big\{ \sum_{\alpha \in \mathcal{I}} \theta_\alpha \phi_\alpha(\mathbf{x}) \big\}$.

In a *minimal* exponential representation, the functions $\{\phi_\alpha\}$ are linearly independent. For example, one minimal representation of a binary process (i.e., $\mathcal{X}_s = \{0, 1\}$ for all s) using pairwise potential functions is the usual Ising model, in which the collection of potentials $\boldsymbol{\phi} = \{x_s \mid s \in V\} \;\cup\; \{\, x_s x_t \mid (s,t) \in E\}$. In this case, the index set is given by $\mathcal{I} = V \cup E$. In most of our analysis, we use an *overcomplete* representation, in which there are linear dependencies among the potentials $\{\phi_\alpha\}$. In particular, we use indicator functions as potentials:

$$\begin{aligned} \phi_{s;j}(x_s) &= \delta_{s;j}(x_s), \quad s \in V;\; j \in \mathcal{X}_s & \text{(1a)} \\ \phi_{st;jk}(x_s, x_t) &= \delta_{s;j}(x_s)\delta_{t;k}(x_t),\; (s,t) \in E;\; (j,k) \in \mathcal{X}_s \times \mathcal{X}_t & \text{(1b)} \end{aligned}$$

where the indicator function $\delta_{s;j}(x_s)$ is equal to one if $x_s = j$, and zero otherwise. In this case, the index set $\mathcal{I}$ consists of the union of $\mathcal{I}(V) = \{\, (s;j) \mid s \in V;\; j \in \mathcal{X}_s\}$ with the edge indices $\mathcal{I}(E) = \{\, (st;jk) \mid (s,t) \in E;\; (j,k) \in \mathcal{X}_s \times \mathcal{X}_t \,\}$.

Of interest to us is the *maximum a posteriori* con£guration $\widehat{\mathbf{x}}_{\mathrm{MAP}} = \arg\max_{\mathbf{x} \in \mathcal{X}^N} p(\mathbf{x}; \theta)$. Equivalently, we can express this MAP con£guration as the solution of the integer program $\mathcal{F}(\theta) = \max_{\mathbf{x} \in \mathcal{X}^N} J(\mathbf{x}; \theta)$, where

$$J(\mathbf{x}; \theta) \;=\; \theta \cdot \boldsymbol{\phi}(\mathbf{x}) \;=\; \sum_{s \in V} \sum_j \theta_{s;j} \phi_{s;j}(x_s) + \sum_{(s,t) \in E} \sum_{j,k} \theta_{st;jk} \phi_{st;jk}(x_s, x_t) \quad (2)$$

Note that the function $\mathcal{F}(\theta)$ is the maximum of a collection of linear functions, and hence is convex [6] as a function of θ, which is a key property for our subsequent development.

1.2 Convex combinations of trees

Let $\widehat{\theta}$ be a particular parameter vector for which we are interested in computing $\mathcal{F}(\widehat{\theta})$. In this section, we show how to derive upper bounds via the convexity of $\mathcal{F}$. Let $\mathcal{T}$ denote a particular spanning tree of G, and let $\mathfrak{T} = \mathfrak{T}(G)$ denote the set of all spanning trees. For each spanning tree $\mathcal{T} \in \mathfrak{T}$, let $\theta(\mathcal{T})$ be an exponential parameter vector of the same dimension as θ that respects the structure of $\mathcal{T}$. To be explicit, if $\mathcal{T}$ is de£ned by an edge set $E(\mathcal{T}) \subset E$, then $\theta(\mathcal{T})$ must have zeros in all elements corresponding to edges[1] not in $E(\mathcal{T})$. For compactness, let $\Theta \triangleq \{\theta(\mathcal{T}) \mid \mathcal{T} \in \mathfrak{T}\}$ denote the full collection, where the notation $\theta(\mathcal{T})$ speci£es those subelements of Θ corresponding to spanning tree $\mathcal{T}$.

In order to de£ne a convex combination, we require a probability distribution $\vec{\boldsymbol{\mu}}$ over the set of spanning trees — that is, a vector $\vec{\boldsymbol{\mu}} \triangleq \{\ \mu(\mathcal{T}),\ \mathcal{T} \in \mathfrak{T} \mid \mu(\mathcal{T}) \geq 0\ \}$ such that $\sum_{\mathcal{T}\in\mathfrak{T}} \mu(\mathcal{T}) = 1$. For any distribution $\vec{\boldsymbol{\mu}}$, we de£ne its *support*, denoted by $\mathrm{supp}(\vec{\boldsymbol{\mu}})$, to be the set of trees to which it assigns strictly positive probability. In the sequel, we will also be interested in the probability $\mu_e = \Pr_{\vec{\boldsymbol{\mu}}}\{e \in \mathcal{T}\}$ that a given edge $e \in E$ appears in a spanning tree $\mathcal{T}$ chosen randomly under $\vec{\boldsymbol{\mu}}$. We let $\boldsymbol{\mu_e} = \{\mu_e \mid e \in E\}$ represent a vector of *edge appearance probabilities*, which must belong to the spanning tree polytope [see 2]. We say that a distribution $\vec{\boldsymbol{\mu}}$ (or the vector $\boldsymbol{\mu_e}$) is *valid* if $\mu_e > 0$ for every edge $e \in E$.

A *convex combination* of exponential parameter vectors is de£ned via the weighted sum $\sum_{\mathcal{T}\in\mathfrak{T}} \mu(\mathcal{T})\theta(\mathcal{T})$, which we denote compactly as $\mathbb{E}_{\vec{\boldsymbol{\mu}}}[\theta(\mathcal{T})]$. Of particular importance are collections of exponential parameters Θ for which there exists a convex combination that is equal to $\widehat{\theta}$. Accordingly, we de£ne the set $\mathcal{A}(\widehat{\theta}) \triangleq \big\{(\Theta; \vec{\boldsymbol{\mu}})\ \big|\ \mathbb{E}_{\vec{\boldsymbol{\mu}}}[\theta(\mathcal{T})] = \widehat{\theta}\big\}$. For any valid distribution $\vec{\boldsymbol{\mu}}$, it can be seen that there exist pairs $(\Theta; \vec{\boldsymbol{\mu}}) \in \mathcal{A}(\widehat{\theta})$.

Example 1 (Single cycle). To illustrate these de£nitions, consider a binary distribution ($\mathcal{X}_s = \{0, 1\}$ for all nodes $s \in V$) de£ned by a single cycle on 4 nodes. Consider a target distribution in the minimal Ising form $p(\mathbf{x}; \widehat{\theta}) = \exp\{x_1x_2 + x_2x_3 + x_3x_4 + x_4x_1 - \Phi(\widehat{\theta})\}$. That is, the target distribution is speci£ed by the minimal parameter $\widehat{\theta} = [0\ 0\ 0\ 0\ 1\ 1\ 1\ 1]$, where the zeros represent the fact that $\widehat{\theta}_s = 0$ for all $s \in V$. The four possible spanning

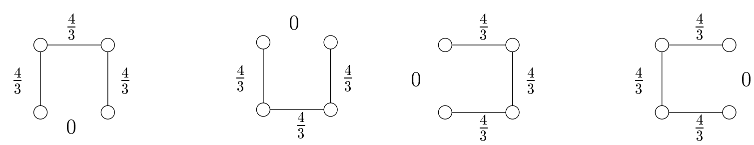


Figure 1. A convex combination of four distributions $p(\mathbf{x}; \theta(\mathcal{T}_i))$, each de£ned by a spanning tree $\mathcal{T}_i$, is used to approximate the target distribution $p(\mathbf{x}; \widehat{\theta})$ on the single-cycle graph.

trees $\mathfrak{T} = \{\mathcal{T}_i \mid i = 1, \ldots, 4\ \}$ on a single cycle on four nodes are illustrated in Figure 1. We de£ne a set of associated exponential parameters $\Theta = \{\theta(\mathcal{T}_i)\}$ as follows:

$$\begin{aligned}
\theta(\mathcal{T}_1) &= \frac{4}{3}[0\ \ 0\ \ 0\ \ 0\ \ 1\ \ 1\ \ 1\ \ 0] &\qquad \theta(\mathcal{T}_3) &= \frac{4}{3}[0\ \ 0\ \ 0\ \ 0\ \ 1\ \ 0\ \ 1\ \ 1] \\
\theta(\mathcal{T}_2) &= \frac{4}{3}[0\ \ 0\ \ 0\ \ 0\ \ 1\ \ 1\ \ 0\ \ 1] &\qquad \theta(\mathcal{T}_4) &= \frac{4}{3}[0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 1\ \ 1\ \ 1]
\end{aligned}$$

Finally, we choose $\mu(\mathcal{T}_i) = 1/4$ for all $\mathcal{T}_i \in \mathfrak{T}$. With this uniform distribution over trees, we have $\mu_e = 3/4$ for each edge, and $\mathbb{E}_{\vec{\boldsymbol{\mu}}}[\theta(\mathcal{T})] = \widehat{\theta}$, so that $(\Theta; \vec{\boldsymbol{\mu}}) \in \mathcal{A}(\widehat{\theta})$.

[1]Given an edge e belonging to two trees $\mathcal{T}_1$ and $\mathcal{T}_2$, the quantity $\theta_e(\mathcal{T}_1)$ can be different than $\theta_e(\mathcal{T}_2)$.

2 Optimal upper bounds

With the set-up of the previous section, the basic form of the upper bounds follows by applying Jensen's inequality [6]. In particular, for any pair $(\Theta; \vec{\boldsymbol{\mu}}) \in \mathcal{A}(\widehat{\theta})$, we have the upper bound $\mathcal{F}(\widehat{\theta}) \leq \mathbb{E}_{\vec{\boldsymbol{\mu}}}[\mathcal{F}(\theta(\mathcal{T}))]$. The goal of this section is to examine this bound, and understand when it is met with equality. In more explicit terms, the upper bound can be written as:

$$\mathcal{F}(\widehat{\theta}) \quad \leq \quad \sum_{\mathcal{T}} \mu(\mathcal{T}) \mathcal{F}(\theta(\mathcal{T})) \;=\; \sum_{\mathcal{T}} \mu(\mathcal{T}) \max_{\mathbf{x} \in \mathcal{X}^N} \big\{ \theta(\mathcal{T}) \cdot \boldsymbol{\phi}(\mathbf{x}) \big\} \tag{3}$$

Now suppose that there exists an $\widehat{\mathbf{x}} \in \mathcal{X}^N$ that attains the maximum de£ning $\mathcal{F}(\theta(\mathcal{T}))$ for each tree $\mathcal{T} \in \operatorname{supp}(\vec{\boldsymbol{\mu}})$. In this case, it is clear that the bound (3) is met with equality. An important implication is that the con£guration $\widehat{\mathbf{x}}$ also attains the maximum de£ning $\mathcal{F}(\widehat{\theta})$, so that it is an *optimal* solution to the original problem.

In fact, as we show below, the converse to this statement also holds. More formally, for any exponential parameter vector $\theta(\mathcal{T})$, let $\operatorname{OPT}(\theta(\mathcal{T}))$ be the collection of con£gurations $\mathbf{x}$ that attain the maximum de£ning $\mathcal{F}(\theta(\mathcal{T}))$. That is

$$\operatorname{OPT}(\theta(\mathcal{T})) \quad = \quad \{\mathbf{x} \in \mathcal{X}^N \mid \theta(\mathcal{T}) \cdot \boldsymbol{\phi}(\mathbf{x}') \;\leq \theta(\mathcal{T}) \cdot \boldsymbol{\phi}(\mathbf{x}) \;\; \text{for all } \; \mathbf{x}' \in \mathcal{X}^N\} \tag{4}$$

With this notation, the critical property is that the intersection $\operatorname{OPT}(\Theta) \triangleq \cap_{\mathcal{T}} \operatorname{OPT}(\theta(\mathcal{T}))$ of con£gurations optimal for all tree-structured problems is non-empty. We thus have the following result:

Proposition 1 (Tightness of bound). *The bound of equation* (3) *is tight if and only if there exists a con£guration* $\widehat{\mathbf{x}} \in \mathcal{X}^N$ *that for each* $\mathcal{T} \in \operatorname{supp}(\vec{\boldsymbol{\mu}})$ *achieves the maximum de£ning* $\mathcal{F}(\theta(\mathcal{T}))$. *(That is,* $\widehat{\mathbf{x}} \in \operatorname{OPT}(\Theta)$.*)*

Proof. Consider some pair $(\Theta; \vec{\boldsymbol{\mu}}) \in \mathcal{A}(\widehat{\theta})$. Let $\widehat{\mathbf{x}}$ be a con£guration that attains the maximum de£ning $\mathcal{F}(\widehat{\theta})$. We write the difference of the RHS and the LHS of equation (3) as follows:

$$\begin{aligned} 0 \quad \leq \quad \Big[\sum_{\mathcal{T}} \mu(\mathcal{T}) \mathcal{F}(\theta(\mathcal{T}))\Big] - \mathcal{F}(\widehat{\theta}) \quad &= \quad \Big[\sum_{\mathcal{T}} \mu(\mathcal{T}) \mathcal{F}(\theta(\mathcal{T}))\Big] - \widehat{\theta} \cdot \boldsymbol{\phi}(\widehat{\mathbf{x}}) \\ &= \quad \sum_{\mathcal{T}} \mu(\mathcal{T}) \big[\mathcal{F}(\theta(\mathcal{T})) - \theta(\mathcal{T}) \cdot \boldsymbol{\phi}(\widehat{\mathbf{x}})\big] \end{aligned}$$

Now for each $\mathcal{T} \in \operatorname{supp}(\vec{\boldsymbol{\mu}})$, the term $\mathcal{F}(\theta(\mathcal{T})) - \theta(\mathcal{T}) \cdot \boldsymbol{\phi}(\widehat{\mathbf{x}})$ is non-negative, and equal to zero only when $\widehat{\mathbf{x}}$ belongs to $\operatorname{OPT}(\theta(\mathcal{T}))$. Therefore, the bound is met with equality if and only if $\widehat{\mathbf{x}}$ achieves the maximum de£ning $\mathcal{F}(\theta(\mathcal{T}))$ for all trees $\mathcal{T} \in \operatorname{supp}(\vec{\boldsymbol{\mu}})$. □

Proposition 1 motivates the following strategy: given a spanning tree distribution $\vec{\boldsymbol{\mu}}$, £nd a collection of exponential parameters $\Theta^* = \{\theta^*(\mathcal{T})\}$ such that the following holds: (a) Admissibility: The pair $(\Theta^*; \vec{\boldsymbol{\mu}})$ satis£es $\sum_{\mathcal{T}} \mu(\mathcal{T}) \theta^*(\mathcal{T}) = \widehat{\theta}$. (b) Mutual agreement: The intersection $\cap_{\mathcal{T}} \operatorname{OPT}(\theta^*(\mathcal{T}))$ of tree-optimal con£gurations is non-empty.

If (for a £xed $\vec{\boldsymbol{\mu}}$) we are able to £nd a collection Θ^* satisfying these two properties, then Proposition 1 guarantees that all con£gurations in the (non-empty) intersection $\cap_{\mathcal{T}} \operatorname{OPT}(\theta^*(\mathcal{T}))$ achieve the maximum de£ning $\mathcal{F}(\widehat{\theta})$. As discussed above, assuming that $\vec{\boldsymbol{\mu}}$ assigns strictly positive probability to every edge in the graph, satisfying the admissibility condition is not dif£cult. It is the second condition of mutual optimality on all trees that poses the challenge.

3 Mutual agreement via equal max-marginals

We now develop an algorithm that attempts to £nd, for a given spanning tree distribution $\vec{\mu}$, a collection $\Theta^* = \{\theta^*(\mathcal{T})\}$ satisfying both of these properties. Interestingly, this algorithm is related to the ordinary max-product algorithm [3, 5], but differs in several key ways. While this algorithm can be formulated in terms of reparameterization [e.g., 5], here we present a set of message-passing updates.

3.1 Max-marginals

The foundation of our development is the fact [1] that any tree-structured distribution $p(\mathbf{x}; \theta(\mathcal{T}))$ can be factored in terms of its max-marginals. In particular, for each node $s \in V$, the corresponding single node max-marginal is de£ned as follows:

$$T_s(x_s) \quad = \quad \max_{\{\mathbf{x}' \mid x'_s = x_s\}} p(\mathbf{x}'; \theta(\mathcal{T})) \tag{5}$$

That is, for each $x_s \in \mathcal{X}_s$, $T_s(x_s)$ is the maximum probability over the subset of con£gurations $\mathbf{x}'$ with element x'_s £xed to x_s. For each edge $(s,t) \in E$, the pairwise max-marginal is de£ned analogously as $T_{st}(x_s, x_t) = \max_{\{\mathbf{x}' \mid (x'_s, x'_t) = (x_s, x_t)\}} p(\mathbf{x}'; \theta(\mathcal{T}))$. With these de£nitions, the max-marginal tree factorization [1] is given by:

$$p(\mathbf{x}; \theta(\mathcal{T})) \quad \propto \quad \prod_{s \in V} T_s(x_s) \prod_{(s,t) \in E(\mathcal{T})} \frac{T_{st}(x_s, x_t)}{T_s(x_s) T_t(x_t)} \tag{6}$$

One interpretation of the ordinary max-product algorithm for trees, as shown in our related work [5], is as computing this alternative representation.

Suppose moreover that for each node $s \in V$, the following uniqueness condition holds:
Uniqueness Condition: *For each $s \in V$, the max-marginal T_s has a unique optimum x_s^*.*
In this case, the vector $\mathbf{x}^* = \{x_s^* \mid s \in V\}$ is the MAP con£guration for the tree-structured distribution [see 5].

3.2 Tree-reweighted max-product

The tree-reweighted max-product method is a message-passing algorithm, with £xed points that specify a collection of tree exponential parameters $\Theta^* = \{\theta^*(\mathcal{T})\}$ satisfying the admissibility condition. The de£ning feature of Θ^* is that the associated tree distributions $p(\mathbf{x}; \theta^*(\mathcal{T}))$ all share a common set $\mathbf{T}^* = \{T_s^*, T_{st}^*\}$ of max-marginals. In particular, for a given tree $\mathcal{T}$ with edge set $E(\mathcal{T})$, the distribution $p(\mathbf{x}; \theta^*(\mathcal{T}))$ is speci£ed compactly by the subcollection $\Pi^{\mathcal{T}}(\mathbf{T}^*) \triangleq \{T_s^* \mid s \in V\} \cup \{T_{st}^* \mid (s,t) \in E(\mathcal{T})\}$ as follows:

$$p(\mathbf{x}; \theta^*(\mathcal{T})) \equiv p^{\mathcal{T}}(\mathbf{x}; \mathbf{T}^*) \;\triangleq\; \kappa \prod_{s \in V} T_s^*(x_s) \prod_{(s,t) \in E(\mathcal{T})} \frac{T_{st}^*(x_s, x_t)}{T_s^*(x_s) T_t^*(x_t)} \tag{7}$$

where κ is a constant[2] independent of $\mathbf{x}$. As long as $\mathbf{T}^*$ satis£es the Uniqueness Condition, the con£guration $\mathbf{x}^* = \{x_s^* \mid s \in V\}$ must be the MAP con£guration for each tree-structured distribution $p(\mathbf{x}; \theta^*(\mathcal{T}))$. This mutual agreement on trees, in conjunction with the admissibility of Θ^*, implies that $\mathbf{x}^*$ is also the MAP con£guration for $p(\mathbf{x}; \widehat{\theta})$.

For each valid $\boldsymbol{\mu_e}$, there exists a tree-reweighted max-product algorithm designed to £nd the requisite set $\mathbf{T}^*$ of max-marginals via a sequence of message-passing operations. For each edge $(s,t) \in E$, let $M_{ts}(x_s)$ be the message passed from node t to node s. It is a vector of length m_s, with one element for each state $j \in \mathcal{X}_s$. We use $\phi_s(x_s; \widehat{\theta})$ as a

[2] We use this notation throughout the paper, where the value of κ may change from line to line.

shorthand for $\sum_j \widehat{\theta}_{s;j}\phi_{s;j}(x_s)$, with the quantity $\phi_{st}(x_s, x_t; \widehat{\theta}_{st})$ similarly de£ned. We use the messages $\mathbf{M} = \{M_{st}\}$ to specify a set of functions $\mathbf{T} = \{T_s, T_{st}\}$ as follows:

$$T_s(x_s) \quad = \quad \exp\left(\phi_s(x_s; \widehat{\theta}_s)\right) \prod_{u \in \Gamma(s)} \left[M_{us}(x_s)\right]^{\mu_{us}} \tag{8a}$$

$$T_{st}(x_s, x_t) \quad = \quad \varphi_{st}(x_s, x_t; \widehat{\theta})\, \frac{\prod_{u \in \Gamma(s)\backslash t} \left[M_{us}(x_s)\right]^{\mu_{us}}}{\left[M_{ts}(x_s)\right]^{(1-\mu_{st})}}\, \frac{\prod_{u \in \Gamma(t)\backslash s} \left[M_{ut}(x_t)\right]^{\mu_{ut}}}{\left[M_{st}(x_t)\right]^{(1-\mu_{ts})}} \tag{8b}$$

where $\varphi_{st}(x_s, x_t; \widehat{\theta}) = \exp\Bigg(\frac{1}{\mu_{st}}\phi_{st}(x_s, x_t; \widehat{\theta}_{st}) + \phi(x_s; \widehat{\theta}_s) + \widehat{\theta}_t\phi(x_t; \widehat{\theta}_t)\Bigg)$.

For each tree $\mathcal{T}$, the subcollection $\Pi^{\mathcal{T}}(\mathbf{T})$ can be used to de£ne a tree-structured distribution $p^{\mathcal{T}}(\mathbf{x}; \mathbf{T})$, in a manner analogous to equation (7). By expanding the expectation $\mathbb{E}_{\vec{\boldsymbol{\mu}}}[\log p^{\mathcal{T}}(\mathbf{x}; \mathbf{T})]$ and making use of the de£nitions of T_s^* and T_{st}^*, we can prove the following:

Lemma 1 (Admissibility). *Given any collection $\{T_s, T_{st}\}$ de£ned by a set of messages $\mathbf{M}$ as in equations* (8a) *and* (8b)*, the convex combination $\sum_{\mathcal{T}} \mu(\mathcal{T}) \log p^{\mathcal{T}}(\mathbf{x}; \mathbf{T})$ is equivalent to $\log p(\mathbf{x}; \widehat{\theta})$ up to an additive constant.*

We now need to ensure that $\mathbf{T} = \{T_s, T_{st}\}$ are a consistent set of max-marginals for each tree-distribution $p^{\mathcal{T}}(\mathbf{x}; \mathbf{T})$. It is suf£cient [1, 5] to impose, for each edge (s, t), the *edgewise consistency* condition $\max_{x_t' \in \mathcal{X}_t} T_{st}(x_s, x_t') = \kappa\, T_s(x_s)$. In order to enforce this condition, we update the messages in the following manner:

Algorithm 1 (Tree reweighted max-product).

1. Initialize the messages $\mathbf{M}^0 = \{M_{st}^0\}$ with arbitrary positive real numbers.

2. For iterations $n = 0, 1, 2, \ldots$, update the messages as follows:

$$M_{ts}^{n+1}(x_s) = \max_{x_t' \in \mathcal{X}_t} \left\{ \exp\left(\frac{1}{\mu_{st}}\phi_{st}(x_s, x_t'; \widehat{\theta}_{st}) + \phi_t(x_t'; \widehat{\theta}_t)\right) \frac{\prod_{u \in \Gamma(t)\backslash s} \left[M_{ut}^n(x_t)\right]^{\mu_{ut}}}{\left[M_{st}^n(x_t)\right]^{(1-\mu_{ts})}} \right\} \tag{9}$$

Using the de£nitions of T_s^* and T_{st}^*, as well as the message update equation (9), the following result can be proved:

Lemma 2 (Edgewise consistency). *Let $\mathbf{M}^*$ be a £xed point of the message update equation* (9)*, and let $\mathbf{T}^* = \{T_s^*, T_{st}^*\}$ be de£ned via $\mathbf{M}^*$ as in equations* (8a) *and* (8b) *respectively. Then the edgewise consistency condition is satis£ed.*

The message update equation (9) is similar to the standard max-product algorithm [3, 5]. Indeed, if G is actually a tree, then we must have $\mu_{st} = 1$ for every edge $(s, t) \in E$, in which case equation (9) is precisely equivalent to the ordinary max-product update. However, if G has cycles, then it is impossible to have $\mu_{st} = 1$ for every edge $(s, t) \in E$, so that the updates in equation (9) differ from ordinary max-product in some key ways. First of all, the weight $\widehat{\theta}_{st}$ on the potential function ϕ_{st} is scaled by the (inverse of the) edge appearance probability $1/\mu_{st} \geq 1$. Secondly, for each neighbor $u \in \Gamma(t)\backslash s$, the incoming message M_{ut} is scaled by the corresponding edge appearance probability $\mu_{ut} \leq 1$. Third of all, in sharp contrast to standard [3] and attenuated [4] max-product updates, the update of message M_{ts} — that is, from t to s along edge (s, t) — depends on the *reverse direction* message M_{st} from s to t along the same edge. Despite these differences, the messages can be updated synchronously as in ordinary max-product. It also possible to perform reparameterization updates over spanning trees, analogous to but distinct from those for ordinary max-product [5]. Such tree-based updates can be terminated once the trees agree on a common con£guration, which may happen prior to message convergence [7].

3.3 Analysis of £xed points

In related work [5], we established the existence of £xed points for the ordinary max-product algorithm for positive compatibility functions on an arbitrary graph. The same proof can be adapted to show that the tree-reweighted max-product algorithm also has at least one £xed point $\mathbf{M}^*$. Any such £xed point $\mathbf{M}^*$ de£nes pseudo-max-marginals $\mathbf{T}^*$ via equations (8a) and (8b), which (by design of the algorithm) have the following property:

Theorem 1 (Exact MAP). *If $\mathbf{T}^*$ satis£es the Uniqueness Condition, then the con£guration $\mathbf{x}^*$ with elements $x_s^* = \arg\max_{x'_s \in \mathcal{X}_s} T_s^*(x'_s)$ is a MAP con£guration for $p(\mathbf{x}; \widehat{\theta})$.*

Proof. For each spanning tree $\mathcal{T} = (V, E(\mathcal{T}))$, the £xed point $\mathbf{T}^*$ de£nes a tree-structured distribution $p(\mathbf{x}; \theta^*(\mathcal{T}))$ via equation (7). By Lemma 2, the elements of $\mathbf{T}^*$ are edgewise consistent. By the equivalence of edgewise and global consistency for trees [1], the subcollection $\Pi^{\mathcal{T}}(\mathbf{T}^*) = \{T_s^* \mid s \in V\} \cup \{T_{st}^* \mid (s,t) \in E(\mathcal{T})\}$ are exact max-marginals for the tree-structured distribution $p(\mathbf{x}; \theta^*(\mathcal{T}))$. As a consequence, the con£guration $\mathbf{x}^*$ must belong to $\mathrm{OPT}(\theta^*(\mathcal{T}))$ for each tree $\mathcal{T}$, so that mutual agreement is satis£ed. By Lemma 1, the convex combination $\mathbb{E}_{\vec{\boldsymbol{\mu}}}[\log p(\mathbf{x}; \theta^*(\mathcal{T}))]$ is equal to $\log p(\mathbf{x}; \widehat{\theta})$, so that admissibility is satis£ed. Proposition 1 then implies that $\mathbf{x}^*$ is a MAP con£guration for $p(\mathbf{x}; \widehat{\theta})$. $\square$

3.4 Failures of tree-reweighted max-product

In all of our experiments so far, the message updates of equation (9), if suitably relaxed, have always converged.[3] Rather than convergence problems, the breakdown of the algorithm appears to stem primarily from failure of the Uniqueness Condition. If this assumption is not satis£ed, we are no longer guaranteed that the mutual agreement condition is satis£ed (i.e., $\mathrm{OPT}(\Theta^*)$ may be empty). Indeed, a con£guration $\mathbf{x}^*$ belongs to $\mathrm{OPT}(\Theta^*)$ if and only if the following conditions hold:

Node optimality: *The element x_s^* must achieve $\max_{x'_s} T_s^*(x'_s)$ for every $s \in V$.*

Edge optimality: *The pair (x_s^*, x_t^*) must achieve $\max_{(x'_s, x'_t)} T_{st}^*(x'_s, x'_t)$ for all $(s,t) \in E$.*

For a given £xed point $\mathbf{T}^*$ that fails the Uniqueness Condition, it may or may not be possible to satisfy these conditions, as the following example illustrates.

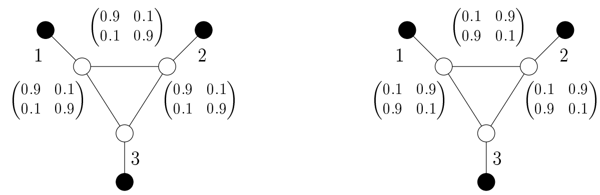
Example 2. Consider the single cycle on three vertices, as illustrated in Figure 2. We de£ne a distribution $p(\mathbf{x}; \widehat{\theta})$ in an indirect manner, by £rst de£ning a set of pseudo-max-marginals $\mathbf{T}^*$ in panel (a). Here $\beta \in [0,1]$ is a parameter to be speci£ed. Observe that the symmetry of this construction ensures that $\mathbf{T}^*$ satis£es the edgewise consistency condition (Lemma 2) for any $\beta \in [0,1]$. For each of the three spanning trees of this graph, the collection $\mathbf{T}^*$ de£nes a tree-structured distribution $p^{\mathcal{T}}(\mathbf{x}; \mathbf{T}^*)$ as in equation (7). We de£ne the underlying distribution via $\log p(\mathbf{x}; \widehat{\theta}) = \mathbb{E}_{\vec{\boldsymbol{\mu}}}[\log p^{\mathcal{T}}(\mathbf{x}; \mathbf{T}^*)] + C$, where $\vec{\boldsymbol{\mu}}$ is the uniform distribution (weight $(1/3)$ on each tree).

In the case $\beta > 0.5$, illustrated in panel (b), it can be seen that two con£gurations — namely $[0\ 0\ 0]$ and $[1\ 1\ 1]$ — satisfy the node and edgewise optimality conditions. Therefore, each of these con£gurations are global maxima for the cost function $\mathbb{E}_{\vec{\boldsymbol{\mu}}}[\log p(\mathbf{x}; \mathbf{T}^*)]$. On the other hand, when $\beta < 0.5$, as illustrated in panel (c), any con£guration $\mathbf{x}^*$ that is edgewise optimal for all three edges must satisfy $x_s^* \neq x_t^*$ for all $(s,t) \in E$. This is clearly impossible, so that the £xed point $\mathbf{T}^*$ cannot be used to specify a MAP assignment.

Of course, it should be recognized that this example was contrived to break down the algorithm. It should also be noted that the standard max-product algorithm can also break

[3] In a relaxed message update, we take an α-step towards the new (log) message, where $\alpha \in (0,1]$ is the step size parameter. To date, we have not been able to prove that relaxed updates will always converge.

$$
\begin{aligned}
T_s^* &= \begin{bmatrix} 1 & 1 \end{bmatrix} \\
T_{st}^* &= \begin{bmatrix} \beta & 1-\beta \\ 1-\beta & \beta \end{bmatrix}
\end{aligned}
$$



(a) (b) (c)

Figure 2. Cases where the Uniqueness Condition fails. (a) Speci£cation of pseudo-max-marginals $\mathbf{T}^*$. (b) For $\beta > 0.5$, both $[0\ 0\ 0]$ and $[1\ 1\ 1]$ are node and edgewise optimal. (c) For $\beta < 0.5$, no con£gurations are node and edgewise optimal on the full graph.

down when this Uniqueness Condition is not satis£ed, as shown in our related work [5].

4 Discussion

This paper demonstrated the utility of convex combinations of tree-structured distributions in upper bounding the log probability of the MAP con£guration. We developed a family of tree-reweighted max-product algorithms for computing optimal upper bounds. In certain cases, the optimal upper bound is met with equality, and hence yields an exact MAP con£guration for the original problem on the graph with cycles. An important open question is to characterize the range of problems for which the upper bound is tight. For problems involving a binary-valued random vector, we have isolated a class of problems for which the upper bound is guaranteed to be tight. We have also investigated the Lagrangian dual associated with the upper bound (3). The dual has a natural interpretation as a tree-relaxed linear program, and has been applied to turbo decoding [7]. Finally, the analysis and upper bounds of this paper can be extended in a straightforward manner to hypertrees of of higher width. In this context, hypertree-reweighted forms of generalized max-product updates [see 5] can again be used to £nd optimal upper bounds, which (when they are tight) again yield exact MAP con£gurations.

References

[1] R. G. Cowell, A. P. Dawid, S. L. Lauritzen, and D. J. Spiegelhalter. *Probablistic Networks and Expert Systems*. Statistics for Engineering and Information Science. Springer-Verlag, 1999.

[2] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. A new class of upper bounds on the log partition function. In *Proc. Uncertainty in Arti£cial Intelligence*, volume 18, pages 536–543, August 2002.

[3] W. T. Freeman and Y. Weiss. On the optimality of solutions of the max-product belief propagation algorithm in arbitrary graphs. *IEEE Trans. Info. Theory*, 47:736–744, 2001.

[4] B. J. Frey and R. Koetter. Exact inference using the attenuated max-product algorithm. In *Advanced mean £eld methods: Theory and Practice*. MIT Press, 2000.

[5] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. Tree consistency and bounds on the max-product algorithm and its generalizations. LIDS Tech. report, MIT; Available online at http://ssg.mit.edu/group/mjwain/mjwain.shtml, July 2002.

[6] D.P. Bertsekas. *Nonlinear programming*. Athena Scienti£c, Belmont, MA, 1995.

[7] J. Feldman, M. J. Wainwright, and D. R. Karger. Linear programming-based decoding and its relation to iterative approaches. In *Proc. Allerton Conf. Comm., Control and Computing*, October 2002.