LETTER Communicated by Alan Yuille

Estimation and Marginalization Using the Kikuchi Approximation Methods

Payam Pakzad
payamp@eecs.berkeley.edu
Venkat Anantharam
ananth@eecs.berkeley.edu
Electrical Engineering and Computer Science, Department, University of California, Berkeley, CA 94720, U.S.A.

In this letter, we examine a general method of approximation, known as the Kikuchi approximation method, for finding the marginals of a product distribution, as well as the corresponding partition function. The Kikuchi approximation method defines a certain constrained optimization problem, called the Kikuchi problem, and treats its stationary points as approximations to the desired marginals. We show how to associate a graph to any Kikuchi problem and describe a class of local message-passing algorithms along the edges of any such graph, which attempt to find the solutions to the problem. Implementation of these algorithms on graphs with fewer edges requires fewer operations in each iteration. We therefore characterize minimal graphs for a Kikuchi problem, which are those with the minimum number of edges. We show with empirical results that these simpler algorithms often offer significant savings in computational complexity, without suffering a loss in the convergence rate. We give conditions for the convexity of a given Kikuchi problem and the exactness of the approximations in terms of the loops of the minimal graph. More precisely, we show that if the minimal graph is cycle free, then the Kikuchi approximation method is exact, and the converse is also true generically. Together with the fact that in the cycle-free case, the iterative algorithms are equivalent to the well-known belief propagation algorithm, our results imply that, generically, the Kikuchi approximation method can be exact if and only if traditional junction tree methods could also solve the problem exactly.

1 Introduction

In its most general form, the problem of finding the marginals of a product distribution is encountered frequently in various branches of science and engineering. An important special case, the probabilistic inference problem (Cowell, Dawid, Lauritzen, & Spiegelhalter 1999), is to infer the most probable scenarios, given a collection of observations. Under a Bayesian

causality model, this is equivalent to finding the marginals of a joint probability distribution in the form of the product of certain conditional probability functions. Applications of the probabilistic inference problem range from medical diagnosis to speech recognition and error-correcting codes.

Example 1. Consider the soft decoding of an (n, k) binary linear code with $(n-k) \times n$ parity check matrix H (see, e.g., Wicker, 1995). Let $P(\mathbf{x}; \mathbf{y}^*)$ represent the joint a posteriori probability density of bits of a code word $\mathbf{x} := (x_1, \ldots, x_n)$, with noisy observations $\mathbf{y}^* := (y_1^*, \ldots, y_n^*)$ over a binary memoryless channel. Then $P(\mathbf{x}; \mathbf{y}^*)$ can be represented as the product of some indicator functions representing the parity checks between the bits of the code words, as well as conditional probabilities representing the noisy observations,

$$P(\mathbf{x}; \mathbf{y}) = \frac{1}{Z} \prod_{i=1}^{n-k} 1\left(\sum_{j=1}^{n} H_{i,j}\, x_j = 0\right) \prod_{j=1}^{n} P(x_i) P(y_i^* | x_i),$$

where the summation is modulo-2, and $1(\cdot)$ is the indicator function, taking values 1 or 0 depending on whether its argument is true or false; Z is a normalizing constant called the partition function. In this case, the marginal $P_i(x_i; \mathbf{y}^*)$ is used to find the most probable value of the ith bit.

In some applications, one is interested mainly in calculating the partition function:

Example 2. In a circuit-switched network, one is interested in finding the invariant distribution of calls in progress along routes of the network. It can be shown (see, e.g., Walrand & Varaiya, 1996) that the invariant distribution has the form

$$\pi(x_1, \ldots, x_M) = \frac{q_1(x_1) \cdots q_M(x_M)}{Z} \prod_{j=1}^{L} 1\left(\sum_{i \in R_j} x_i < n_j\right).$$

Here, M is the total number of routes, x_i is the number of calls along route i, $q_i(x_i)$ is the (known) invariant distribution of x_i if the links had an infinite number of circuits, L is the number of links in the network, n_j is the capacity of link j, and $R_j \subset \{1, \ldots, M\}$ is the index set of routes that use link j. Finally, Z is the partition function, defined by

$$Z := \sum_{x_1, \ldots, x_M} \prod_{i=1}^{M} q_i(x_i) \prod_{j=1}^{L} 1\left(\sum_{i \in R_j} x_i < n_j\right).$$

Therefore, in order to calculate the invariant distribution, one needs only to calculate the partition function Z.

As another example, in physics, one can derive various thermodynamical properties of a system, such as the average energy and entropy, if the partition function is known as a function of the temperature (see, e.g., Kittel & Kroemer, 1980).

Although the general marginalization problem can be exponentially complex, scientists and engineers have long explored ways to reduce the computational complexity of the calculations required to find the marginals, either exactly or approximately, (see, e.g., Pearl, 1988; Aji & McEliece, 2000; Morita, 1994; Yedidia, Freeman, & Weiss 2001; Luby, 2002; Pakzad & Anantharam, 2004). Most approaches use a graphical model to represent the interdependence of variables in the factor functions and use message-passing algorithms on this graph to localize the calculations. Belief propagation (Pearl, 1988) is one such algorithm. The success of low-density parity check (LDPC) codes (Gallager, 1963; MacKay & Neal, 1995) and turbo codes (Berrou, Glavieux, & Thitimajshima, 1993), which are decoded using instances of the belief propagation algorithm on a loopy graph (McEliece, MacKay, & Cheng, 1998), motivated many communications engineers to look more closely at belief propagation and junction graphs. So far, however, a general characterization of the quality of approximation and convergence properties of loopy belief propagation has not been discovered, despite a number of excellent partial results, which have considerably increased our understanding of the dynamics of such algorithms (see, e.g., Richardson & Urbanke, 2001; Weiss, 2000; Richardson, Shokrollahi, & Urbanke, 2001; Divsalar, Jin, & McEliece, 1998; Richardson, 2000; MacKay & Neal, 1995).

Yedidia et al. (2001) showed recently that there is a close connection between loopy belief propagation and certain approximations to the variational free energy in statistical physics. Specifically, as we will also discuss in this article, the fixed points of the belief propagation algorithm were shown to coincide with the stationary points of the Bethe free energy subject to consistency constraints. Here, the Bethe free energy is an approximation to the variational free energy. The Bethe approximation is only a special case of a more general class of approximations called the Kikuchi approximations (Kikuchi, 1951). A class of iterative message-passing algorithms was introduced in Yedidia et al. (2001), which attempt to find the stationary points of the Kikuchi free energy. Using such message-passing algorithms, one is expected to obtain better approximations to the marginals and the partition function than the ones given by loopy belief propagation.

Building on the generalized region-based approach introduced in Pakzad and Anantharam (2002a, 2002b), in this article, we explore a wide range of ideas related to the Kikuchi approximation method. In particular, we discuss necessary conditions for uniqueness of the minimizers of the Kikuchi free energy, introduce graphical representations for the problem, and define

minimal graphical representations, which result in iterative solutions that are often significantly less complex than the algorithms discussed in Yedidia, Freeman, and Weiss (2001, 2002), and McEliece and Yildrim (2003). Furthermore, we will show that for generic problems, the Kikuchi approximation yields the exact marginals if and only if this minimal graphical representation of the Kikuchi problem is loop free.[1] We will also address the more general problem of approximating the entropy of a product distribution in terms of the entropies of its marginals.

Yedidia et al. (2002) independently developed a similar general framework based on region graphs. As such, there is some intersection between our work and theirs; in particular, the presentation in sections 2 and 3 should be compared with those in Yedidia et al. (2002), as well as Aji and McEliece (2001) and McEliece and Yildrim (2003). However, the major contributions of this work—the discussion of the graphical representations of Kikuchi problems, the convexity conditions, the necessary and sufficient conditions for exactness of the approximation, and the low-complexity extension of the generalized belief propagation algorithm—have not been previously addressed. We point out these similarities and differences throughout the article.

Other researchers have developed various techniques based on related ideas, each with specific advantages over traditional loopy belief propagation. Yuille (2002) derives a double-loop, free-energy-minimizing algorithm that is guaranteed to converge, unlike loopy belief propagation. Welling and Teh (2001) formulate an algorithm of gradient-descent type, which is guaranteed to find a fixed point of the Bethe free energy. Wainwright and Jordan (2003) discuss convex relaxations of the variational principle, resulting in efficient algorithms that yield upper bounds to the partition function.

The outline of this letter is as follows. We define the marginalization problem and set up some necessary notation in section 2. In section 3, we review the connection with methods in statistical physics, define the Kikuchi approximation method as one that approximates the desired marginals as the constrained fixed points of an appropriately defined free energy functional, and show that there are iterative message-passing algorithms whose fixed points correspond to the stationary points of the Kikuchi functional. Sufficient conditions for convexity of the Kikuchi functional are also provided. The restriction of these conditions to the Bethe case implies the well-known result on the convergence of loopy belief propagation on graphs with a single loop (see, e.g., Weiss, 2000).

In section 4 we introduce the notion of graphical representations for a Kikuchi problem, establish the connection with junction trees, and prove results on the exactness of the Kikuchi approximation. In section 5 we derive

[1] By Kikuchi problem, we mean the problem of minimizing the Kikuchi free energy, subject to some consistency constraints.

the generalized belief propagation (GBP) algorithm of Yedidia et al. (2001) on any arbitrary graphical representation of a Kikuchi problem. This is an extension of the results in Yedidia et al. (2002) and McEliece and Yildrim (2003). Interested readers are referred to Pakzad (2004) for further technical discussions of these topics. Some experimental results are reported in section 6, comparing the convergence properties of the low-complexity GBP algorithm derived here, with the algorithms described in Yedidia et al. (2002) and McEliece and Yildrim (2003).

2 Problem Setup

Let $\mathbf{x} := (x_0, \ldots, x_{N-1})$, where for each $i \in [N] := \{0, \ldots, N-1\}$, x_i is a variable taking value in $[q_i] := \{0, \ldots, q_i - 1\}$, with $q_i \geq 2$.

Let R be a collection of subsets of $[N]$; we call each $r \in R$ a *region*. We assume that each variable index $i \in [N]$ appears in at least one region $r \in R$.

Associated with each region $r \in R$ is a nonnegative kernel function, $\alpha_r(\mathbf{x}_r)$, depending only on the variables that appear in r. Then the corresponding R-decomposable (Boltzmann) product distribution is defined as

$$B(\mathbf{x}) := \frac{1}{Z} \prod_{r \in R} \alpha_r(\mathbf{x}_r). \tag{2.1}$$

Here Z is the normalizing constant and is the *partition function*. For a subset $s \subset [N]$, we denote by $B_s(\mathbf{x}_s) := \sum_{\mathbf{x}_{[N]\setminus s}} B(\mathbf{x})$ the s-marginal of $B(\mathbf{x})$.

Problem. The problem considered in this letter is that of finding one or more of the $B_r(\mathbf{x}_r)$'s for $r \in R$, and/or the partition function Z.

The methods developed in this article to solve this problem are best described in the language of partially ordered sets or posets (see, e.g., Stanley, 1986). Specifically, the collection R of regions can be viewed as a poset with set inclusion as its partial ordering relation. This is because inclusion is reflexive ($\forall r \in R, r \subseteq r$), antisymmetrical ($r \subseteq s$ and $s \subseteq r$ implies $r = s$), and transitive ($r \subseteq s$ and $s \subseteq t$ implies $r \subseteq t$). We write $r \subset t$ to denote strict inclusion. We say t *covers* u in R, and write $u \prec t$, if $u, t \in R$, $u \subset t$ and $\nexists v \in R$ s.t. $u \subset v \subset t$.

Definition 1. *Given a poset R, its Hasse diagram G_R is a directed acyclic graph (DAG)[2] whose vertices are the elements of R and whose edges correspond to cover relations in R; that is, an edge $(t \to u)$ exists in G_R iff $u \prec t$.*

[2] Traditionally the Hasse diagram is drawn as an undirected graph, with an implied upward direction (see Stanley, 1986). This is indeed equivalent to a DAG, which will be the view used in this letter.

It follows that for any two distinct nodes $r, s \in R$, we have $r \subset s$ iff there is a directed path from s to r in G_R.

Throughout this letter, we will need the following definitions. Let R be a poset of subsets of $[N]$ with the partial ordering of inclusion. For each subset $r \subseteq [N]$, we define:

$$\begin{aligned} &\text{Ancestors:} && \mathcal{A}(r) := \{s \in R \,:\, r \subset s\} \\ &\text{Descendants:} && \mathcal{D}(r) := \{s \in R \,:\, s \subset r\} \\ &\text{Forebears:} && \mathcal{F}(r) := \{s \in R \,:\, r \subseteq s\} \end{aligned}$$

Further, for $r \in R$ we define:

$$\begin{aligned} &\text{Parents:} && \mathcal{P}(r) := \{s \in R : r \prec s\} \\ &\text{Children:} && \mathcal{C}(r) := \{s \in R : s \prec r\} \end{aligned}$$

Note that in each of these definitions, the collection of subsets being defined comprises regions, even though the argument r of $\mathcal{A}(r)$, $\mathcal{D}(r)$, and $\mathcal{F}(r)$ need not be a region itself. For a collection S of subsets of $[N]$, we define $\mathcal{F}(S) := \bigcup_{s \in S} \mathcal{F}(s)$. A subset T of R that is in the form $T = \mathcal{F}(S)$ for some $S \subseteq R$, viewed as a sub-poset of R (with the same partial ordering), is called an *up-set* of R. Finally we define the *depth* of each region $r \in R$ as:

$$d(r) := \begin{cases} 0 & \text{if } r \text{ is maximal} \\ 1 + \max_{s \in \mathcal{P}(r)} d(s) & \text{otherwise.} \end{cases}$$

3 Kikuchi Approximation Method

3.1 Connection with Statistical Physics. In the setup described in section 2, we can view x_i as the spin of the particle at position i in a system of N particles. Let $b(\mathbf{x})$ denote a probability distribution on the configuration of spins, and consider a function $E(\mathbf{x})$ called the energy function. Suppose the energy function is R-decomposable, that is, $E(\mathbf{x}) = \sum_{r \in R} E_r(\mathbf{x}_r)$ for certain functions $\{E_r(\mathbf{x}_r), r \in R\}$.

In statistical physics, one defines the (Helmholtz) *variational free energy* as the following functional of the distribution

$$F(b(\mathbf{x})) := U(b(\mathbf{x})) - H(b(\mathbf{x})), \tag{3.1}$$

where $U := \sum_{\mathbf{x}} b(\mathbf{x}) E(\mathbf{x})$ is the average energy and $H := -\sum_{\mathbf{x}} b(\mathbf{x}) \log(b(\mathbf{x}))$ is the entropy of the system. We make the connection with the problem

formulation of section 2 by setting $E_r(\mathbf{x}_r) := -\log(\alpha_r(\mathbf{x}_r))$. We can then write

$$\begin{aligned} E(x) &= \sum_{r\in R} E_r(\mathbf{x}_r) \\ &= -\sum_{r\in R} \log(\alpha_r(\mathbf{x}_r)) \\ &= -\log\left(\frac{1}{Z}\prod_{r\in R}\alpha_r(\mathbf{x}_r)\right) - \log(Z) \\ &= -\log(B(\mathbf{x})) - \log(Z), \end{aligned}$$

where $B(\mathbf{x})$ is the Boltzmann distribution of equation 2.1. Then the variational free energy can be rewritten as follows:

$$\begin{aligned} F(b) &= \sum_{\mathbf{x}} b(\mathbf{x})(-\log(B(\mathbf{x})) - \log(Z)) + \sum_{\mathbf{x}} b(\mathbf{x})\log(b(\mathbf{x})) \\ &= \sum_{\mathbf{x}} b(\mathbf{x})\log\left(\frac{b(\mathbf{x})}{B(\mathbf{x})}\right) - \log(Z) \\ &= \mathrm{KL}(b||B) - \log(Z), \end{aligned}$$

where $\mathrm{KL}(b||B)$ is the Kullback-Leibler divergence between $b(\mathbf{x})$ and $B(\mathbf{x})$ (see, e.g., Cover & Thomas, 1991). It is then clear that $F(b)$ is uniquely minimized when $b(\mathbf{x})$ equals the Boltzmann distribution $B(\mathbf{x})$ of equation 2.1, and we have

$$F_0 := \min_{b(\mathbf{x})} F(b(\mathbf{x})) = F(B(\mathbf{x})) = -\log(Z). \tag{3.2}$$

As mentioned in section 1, equation 3.2 is of great interest in science and engineering. Physicists are interested in finding the log-partition function F_0, as a function of a temperature variable, which we have omitted here, since thermodynamical properties of physical systems can be derived from it. In estimation problems in engineering, one is interested in finding the marginals of the Boltzmann distribution $B(\mathbf{x})$. This is called the probabilistic inference problem. However, equation 3.2, viewed as an optimization problem, does not prescribe a practical way for computing these quantities, as it involves minimization over the exponentially large domain of distributions $b(\mathbf{x})$.

Given that the energy function is R-decomposable, to simplify the minimization problem 3.2, one may try to reformulate it in a way that is, loosely speaking, also R-decomposable. A natural way to do this is to try to represent the free energy as a functional of the R-marginals of the distribution $b(\mathbf{x})$.

Definition 2. *We will call a collection* $\{b_r(\mathbf{x}_r), r \in R\}$ *of probability functions, which may or may not be the marginals of a single distribution, a collection of*

R-pseudomarginals. A collection of R-pseudomarginals that are also the marginals of a probability distribution $b(\mathbf{x})$ are called the R-marginals of $b(\mathbf{x})$.

Define Δ_R to be the family of the R-marginals of all probability distributions on $\mathbf{x}$; that is, a collection $\{b_r(\mathbf{x}_r), r \in R\}$ belongs to Δ_R if and only if there exists a distribution $b(\mathbf{x})$ s.t. $\forall r \in R,\ b_r(\mathbf{x}_r) = \sum_{\mathbf{x}_{[N]\setminus r}} b(\mathbf{x})$. Then we can rewrite equation 3.2 as

$$\begin{aligned} F_0 &= \min_{\{b_r(\mathbf{x}_r)\}\in\Delta_R} F_R(\{b_r(\mathbf{x}_r)\}) \\ \{b_r^*(\mathbf{x}_r)\} &= \arg\min_{\{b_r(\mathbf{x}_r)\}\in\Delta_R} F_R(\{b_r(\mathbf{x}_r)\}) \end{aligned} \tag{3.3}$$

where

$$F_R(\{b_r\}) := \min_{b(\mathbf{x}):\{b_r\}\ R\text{-marginals of }b} F(b(\mathbf{x})).$$

Since $E(\mathbf{x})$ is R-decomposable, the average energy decomposes as

$$U(b(\mathbf{x})) = \sum_{r\in R}\sum_{\mathbf{x}_r} b_r(\mathbf{x}_r)E_r(\mathbf{x}_r), \tag{3.4}$$

where $b_r(\mathbf{x}_r)$'s are the marginals of distribution $b(\mathbf{x})$. In general, however, the entropy term in the free energy, equation 3.1, cannot be decomposed in terms of the R-marginals of $b(\mathbf{x})$. The key component of the Kikuchi approximation method is to use an approximation of the form

$$H(b(\mathbf{x})) \simeq \sum_{r\in R} k_r\, H_r(b_r(\mathbf{x}_r)), \tag{3.5}$$

where $H_r(b_r(\mathbf{x}_r)) := -\sum_{\mathbf{x}_r} b_r(\mathbf{x}_r)\log(b_r(\mathbf{x}_r))$ is the regional entropy associated with a region $r \in R$, and k_r's are suitable constants to be determined.

As discussed in Pakzad and Anantharam (2002a), we may view $R \cup \{[N]\}$ as a poset with partial ordering of inclusion. For each $r \in R$, define $c_r := -\mu(r, [N])$ where $\mu(\cdot,\cdot)$ is the Möbius function. Then the Möbius inversion formula (see, e.g., Stanley, 1986) shows that c_r's are defined uniquely by the following equations,

$$c_r = 1 - \sum_{s\in\mathcal{A}(r)} c_s, \tag{3.6}$$

where $\mathcal{A}(r)$ is the set of ancestors of r, as defined in section 2. Following Yedidia et al. (2001), we call the c_r's defined in this manner the *overcounting factors*. As it turns out, c_r's are the natural choice for the constants $\{k_r\}$

in equation 3.5, as we show in proposition 1 (see Pakzad, 2004, for the proof).

Proposition 1. *The only choice of factors $\{k_r\}$ that can result in exactness of equation 3.5 for all R-decomposable Boltzmann distributions—that is, distributions $b(\mathbf{x}) := \frac{1}{Z}\prod_{r\in R}\alpha_r(\mathbf{x}_r)$ for all choices of $\{\alpha_r, r \in R\}$—is the overcounting factors $\{c_r\}$.*

In fact, the original choice of $\{k_r\}$ in the Kikuchi approximation method (Kikuchi, 1951) was also $\{k_r\} = \{c_r\}$. It will also be shown that this exactness happens if and only the collection R of regions is loopfree in an appropriate sense, which will be defined in section 4.1.

The Kikuchi approximation method, which will be defined more formally in section 3.2, proposes to solve a constrained minimization problem of the following form (cf. equation 3.3),

$$\{B_r(\mathbf{x}_r)\} \simeq \{b_r^*(\mathbf{x}_r)\} := \arg \min_{\{b_r(\mathbf{x}_r)\}\in\Delta_R^K} F_R^K(\{b_r(\mathbf{x}_r)\}). \tag{3.7}$$

Here $F_R^K(\{b_r\})$, known as the Kikuchi free energy[3] (see, e.g., Kikuchi, 1951), is defined as follows,

$$F_R^K(\{b_r(\mathbf{x}_r)\}) := \sum_{r\in R}\sum_{\mathbf{x}_r} b_r(\mathbf{x}_r)E_r(\mathbf{x}_r) + \sum_{r\in R}\sum_{\mathbf{x}_r} c_r\, b_r(\mathbf{x}_r)\log(b_r(\mathbf{x}_r)), \tag{3.8}$$

and Δ_R^K is a set of constraints to enforce consistency between the b_r's, defined as

$$\Delta_R^K := \left\{ \{b_r(\mathbf{x}_r),\ r \in R\} \ :\ \forall t, u \in R \ \text{ s.t. } t \subset u, \sum_{\mathbf{x}_{u\setminus t}} b_u(\mathbf{x}_u) = b_t(\mathbf{x}_t) \right.$$
$$\left. \text{and } \ \forall u \in R, \ \sum_{\mathbf{x}_u} b_u(\mathbf{x}_u) = 1 \right\}. \tag{3.9}$$

Note that in general, the constraints of Δ_R^K are not enough to guarantee that every collection of pseudomarginals $\{b_r,\ r \in R\} \in \Delta_R^K$ is in fact the collection of the marginals of a single distribution function $b(\mathbf{x})$. A collection may very well satisfy all the consistency constraints of equation 3.9 and not be the marginals of any distribution.

In section 4 we discuss conditions on R that guarantee that the free energy $F(b)$ can be viewed as a functional of the marginals of $b(\mathbf{x})$, that is, $\{b_r, r \in R\}$,

[3] Yedidia et al. (2002) call this functional the region graph free energy.

and, as such a functional, equals the Kikuchi functional F_R^K. Further, we discuss conditions on R under which the constraint set Δ_R^K equals the family of R-marginals Δ_R.

3.2 Kikuchi Approximation Method. In this section we formulate the Kikuchi approximation method for solving the marginalization problem posed in section 2. We will further describe conditions on the collection of regions R, which are expected to improve the quality of the approximations.

Let R_0 be a collection of regions and $\{\alpha_r^0(\mathbf{x}_r),\ r \in R_0\}$ be a collection of kernel functions. We are interested in solving the marginalization problem posed in section 2 for R_0 and $\{\alpha_r^0\}$.

Let R be another collection of regions obtained from R_0 in such a way that $\forall r' \in R_0$, $\exists r \in R$ s.t. $r' \subseteq r$. Then one can always form[4] a collection of R-kernels $\{\alpha_r(\mathbf{x}_r),\ r \in R\}$ so that $-\sum_{r\in R}\log(\alpha_r(\mathbf{x}_r)) = -\sum_{r\in R_0}\log(\alpha_r^0(\mathbf{x}_r)) =: E(\mathbf{x})$.

Now for each $r \in R$, define $\beta_r(\mathbf{x}_r) := \prod_{s\subseteq r}\alpha_s(\mathbf{x}_s)$. Then the Boltzmann distribution of equation 2.1 takes the following product forms,

$$B(\mathbf{x}) = \frac{\prod_{r'\in R_0}\alpha_{r'}^0(\mathbf{x}_{r'})}{Z} = \frac{\prod_{r\in R}\alpha_r(\mathbf{x}_r)}{Z} = \frac{\prod_{r\in R}\beta_r(\mathbf{x}_r)^{c_r}}{Z}, \tag{3.10}$$

where the last equality follows from the fact that by equation 3.6, $\sum_{r\in\mathcal{F}(s)} c_r = 1$ for all $s \in R$. Using approximations 3.8 and 3.9, we are now interested in solving the following:

Problem (Kikuchi Approximation).

$$\begin{aligned} -\log(Z) \simeq F^* &:= \min_{\{b_r(\mathbf{x}_r)\}\in\Delta_R^K} F_R^K(\{b_r(\mathbf{x}_r)\}) \\ \text{and}\quad \{B_r(\mathbf{x}_r)\} \simeq \{b_r^*(\mathbf{x}_r)\} &:= \arg\min_{\{b_r(\mathbf{x}_r)\}\in\Delta_R^K} F_R^K(\{b_r(\mathbf{x}_r)\}). \end{aligned} \tag{3.11}$$

Note now that by equation 3.3, if $F(b(\mathbf{x})) = F_R^K(\{b_r(\mathbf{x}_r)\})$ for all $b(\mathbf{x})$, and $\Delta_R = \Delta_R^K$, then the minimizer collection $\{b_r^*(\mathbf{x}_r)\}$ of equation 3.11 would correspond exactly to the collection of the marginals of the product function $B(\mathbf{x})$ of equation 3.10. Hence, if the Kikuchi approximate free energy $F_R^K(\{b_r\})$ is close to $F(b)$, and local consistency constraint set Δ_R^K is also close to Δ_R, the minimizers $\{b_r^*\}$ of equation 3.11 are expected to be close approximations to these marginals.

Our focus in the rest of this article shifts to the above problem and to the relation between the solution to this problem and the original one in section 2. An important question, which we address in detail, is when the

[4] Note however that the way this assignment is done can have an impact on the quality of the approximations to equation 3.3 provided by equation 3.11.

b_r^*'s are equal to the marginals B_r of the Boltzmann distribution. We also address in detail in sections 4 and 5 message-passing algorithms on graphs that solve equation 3.11. These algorithms are similar in nature to the GBP of Yedidia et al. (2002) and poset belief propagation of McEliece and Yildrim (2003), although as we will show, the extensions presented here are often considerably more efficient than those algorithms.

The collection R of regions effectively specifies both the Kikuchi approximation, equation 3.8, and the constraint set, equation 3.9. It is also evident that equation 3.11 as an approximation method can be applied for any given F_R^K and Δ_R^K; better choices of R simply result in better approximations. Therefore, we can define the Kikuchi approximation method as the general class of constrained minimization problems given by equation 3.11, which are parameterized by the poset[5] R of regions, and local kernel functions $\alpha_r(\mathbf{x}_r)$ for each $r \in R$.

It remains to specify which choices of R yield good approximations of the marginals. In the remainder of this article, we consider only collections of regions R that have the same maximal regions as R_0. Expansion of the maximal regions corresponds to clustering methods as discussed in Pearl (1988). The techniques developed here to derive low-complexity message-passing algorithms to solve the Kikuchi approximation problem can also be applied after clustering.

It certainly seems that minimization with more local consistency constraints on $\{b_r(\mathbf{x}_r)\}$ should result in better approximations, since the true marginals would satisfy all such constraints. At the same time, the entropy approximations of the type given in equation 3.5 are also expected to improve if more regions are included. Therefore, one might conclude that for a given collection of maximal regions of R_0, augmenting them by introducing additional subregions to form R, where the α_r's corresponding to the augmented subregions are taken to be 1, should improve the approximation (at the expense of increasing the complexity of the underlying minimization).

Let G be a labeled graph whose vertices are identified with subsets of $[N]$. We define the following *connectivity conditions* on G:

A1: $\forall\, i \in [N]$, the subgraph of G consisting of the regions in $\mathcal{F}(\{i\})$ is connected.

Generalizing this, we can devise condition (**A***n*) on G, for each $n \in \{1, \ldots, N\}$ as follows:

A*n:* $\forall\, s \subset [N]$, $|s| \leq n$, the subgraph of G on regions in $\mathcal{F}(s)$ is connected.

[5] Note that although inclusion is certainly the most natural partial ordering for R, the problem is well defined for any arbitrary partial ordering.

We say a poset R has property $\mathbf{A}n$ iff its Hasse diagram G_R satisfies condition $\mathbf{A}n$. Note that in the context of the Kikuchi problem, equation 3.11, property $\mathbf{A}n$ guarantees that the beliefs at all regions will be consistent at the level of any subset $\mathbf{x}_r$ of the variables of cardinality up to n. It is therefore natural to require that R satisfies at least condition $\mathbf{A}1$. We call a poset R satisfying $\mathbf{A}n$ for all n a *totally connected* poset.

Inspired by Aji and McEliece (2001), one might insist that acceptable approximations of the entropy term 3.5 are those in which each variable x_i appears the same number of times on the two sides of the equality sign:

$$\mathbf{B}1\text{:} \sum_{r \in \mathcal{F}(\{i\})} c_r = 1 \quad \text{for each} \quad i = 0, \ldots, N-1.$$

We can extend this condition as follows:

$$\mathbf{B}n\text{:} \sum_{r \in \mathcal{F}(s)} c_r = 1 \quad \text{for each} \quad s \subset [N], \;\; |s| \leq n \;\; \text{s.t.} \;\; \mathcal{F}(s) \neq \emptyset.$$

Conditions $\mathbf{B}n$ are called the *balance conditions*, and we call a poset R satisfying $\mathbf{B}n$ for all n, a *totally balanced* poset.

These conditions are expected to give progressively better approximate solutions, although they will not in general guarantee an exact solution.

The original cluster variation method of Kikuchi as defined in Morita (1994) and Yedidia et al. (2001) in effect chooses R to be the smallest collection of regions including R_0, which is closed under nonempty intersection of regions. The following proposition shows that the choice of R made in the cluster variation method is expected to give a reasonable Kikuchi approximation.

Proposition 2. *Any collection of regions R that is closed under nonempty intersection of regions is totally connected and totally balanced.*

Proof. See Pakzad (2004).

The special case when the Hasse diagram G_R has depth 2 (i.e., there are no distinct $r, s, t \in R$ such that $r \subset s \subset t$) is called the *Bethe case* in this article. In this case, G_R can be viewed as a hypergraph in which the maximal regions of R are the vertices and the minimal regions are the hyperedges. Aji and McEliece (2001) consider the case when the hypergraph view of G_R is a graph, that is, the minimal elements of R are covered by at most two regions, so the hyperedges are in fact edges. It can be immediately verified that the junction graph condition given in Aji and McEliece (2001) is simply the intersection of conditions $\mathbf{A}1$ and $\mathbf{B}1$ above. On the other

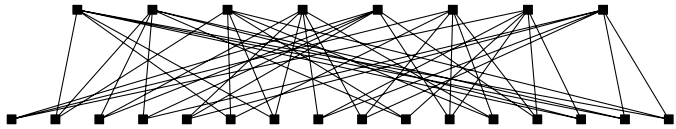

Figure 1: Tanner graph of a linear code.

hand, the generalized notion of junction graphs defined in Yedidia et al. (2002) is essentially the same as our Bethe case, with the minor exception that junction graphs are defined to satisfy condtition **B**1, whereas we do not a priori require that.

We now give an example to illustrate some of the notions defined in this section.

Example 3. Consider the $(16, 8)$ linear code represented by the bipartite graph of Figure 1, where the top nodes correspond to parity checks and the bottom nodes correspond to symbol bits. This graph can be interpreted as the Hasse diagram of a two-level poset, where the regions associated with the bit nodes are $\{1\}, \{2\}, \ldots, \{16\}$, respectively, and the region associated with each check node is the subset of $\{1, \ldots, 16\}$, corresponding to the bits that constitute that parity check. This is an example of the Bethe case, where the regions corresponding to the check nodes are maximal and those corresponding to the bit nodes are minimal. The overcounting factors corresponding to the check nodes are equal to 1, while those corresponding to the bit nodes equal "one minus the number of check nodes connected to that bit node." In this case each bit node is connected to three check nodes, so that the overcounting factors for all bit nodes equal $1 - 3 = -2$. In this case, the GBP algorithm we discuss in section 5 will reduce to the original Gallager-Tanner decoding algorithm for LDPC codes (see Gallager, 1963; Tanner, 1981).

This poset has property **A**1 but not **A**2: for example, the regions corresponding to the first and third check nodes are $\{2, 6, 7, 14, 15, 16\}$ and $\{2, 6, 7, 10, 12, 16\}$, respectively, both containing $\{2, 6\}$, but they are not connected through regions that contain $\{2, 6\}$.

Also, this poset satisfies **B**1 but not **B**2: with $s := \{2, 6\}$, $\mathcal{F}(s)$ is precisely the first and third check-node regions, and $\sum_{r \in \mathcal{F}(s)} c_r = 1 + 1 = 2 \neq 1$.

On the other hand, one can throw in all the intersections of the check node regions to create the poset whose Hasse diagram is shown in Figure 2.

Here, the nodes in the middle row correspond to the intersections of the check node regions, in the first row, to which they are connected; for example, the second node in the middle row corresponds to region $\{2, 6, 7, 16\}$, which is the intersection of the first and third check node regions.

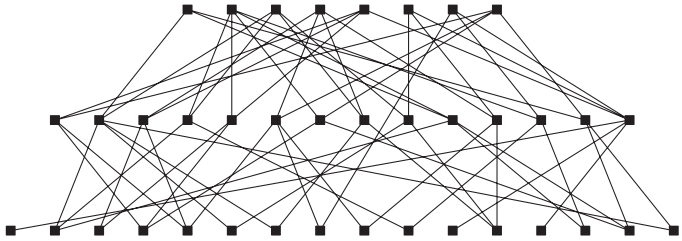

Figure 2: Alternative poset for the linear code of example 3.

It is easy to verify that this poset is totally connected and totally balanced.

3.3 Lagrange Multipliers and Iterative Solutions. Lagrange's method can be used to solve the constrained minimization problem 3.11. We form the Lagrangian:

$$\begin{aligned}\mathcal{L} := &\sum_{r\in R}\sum_{\mathbf{x}_r}(-b_r(\mathbf{x}_r)\log(\alpha_r(\mathbf{x}_r)) + c_r\, b_r(\mathbf{x}_r)\log(b_r(\mathbf{x}_r)))\\ &+ \sum_{r\in R}\sum_{t\prec r}\sum_{\mathbf{x}_t}\lambda_{rt}(\mathbf{x}_t)\left(b_t(\mathbf{x}_t) - \sum_{\mathbf{x}_{r\setminus t}} b_r(\mathbf{x}_r)\right)\\ &+ \sum_{r\in R}\kappa_r\left(\sum_{\mathbf{x}_r} b_r(\mathbf{x}_r) - 1\right),\end{aligned} \tag{3.12}$$

where coefficients $\lambda_{rt}(s_t)$ enforce consistency constraints and coefficients κ_r enforce normalization constraints, and as before $t \prec r$ means that r covers t. Note that since the edge constraints of G_R are a sufficient representation of Δ_R^K, as discussed before, we need only define λ_{rt} for pairs $r, t \in R$ with $t \prec r$, that is, along the edges of G_R.

Setting partial derivative $\partial\mathcal{L}/\partial b_r(\mathbf{x}_r) = 0$ for each $r \in R$ gives an equation for $b_r(\mathbf{x}_r)$ in terms of λ_{ur}'s and λ_{rt}'s. The consistency constraints give update rules for each λ_{rt} in terms of other λ multipliers. Once a set of messages m_{rt} (from r to t, for each edge $(r \to t)$ of G_R) has been defined in terms of the Lagrange multipliers λ_{rt}'s, these update rules define an iterative algorithm whose fixed points are the stationary points of the given constrained minimization problem.

In section 5 we will give a detailed derivation for an important algorithm of this type, called the generalized belief propagation (GBP) algorithm (see also Yedidia et al., 2001), and we will also see that the belief propagation algorithm of Pearl (1988) is the restriction of the above algorithm in the Bethe case. The version of GBP algorithm presented here is an extension of the one originally introduced in Yedidia et al. (2001), in which we take advantage of certain systematic complexity-reducing transformations that will be

described in section 4. The resulting algorithm can be considerably less complex than the original GBP of Yedidia et al. (2001) and its generalizations in Yedidia et al. (2002) and McEliece and Yildrim (2003), constituting a major practical contribution of this work.

3.4 Convexity Conditions. In this section we describe our results regarding the convexity of the optimization problem 3.11, which we first reported in Pakzad and Anantharam (2002a).

The Kikuchi free energy (see equation 3.8) constrained on $\{b_r\} \in \Delta_R^K$ is bounded below, and hence the constrained minimization problem of equation 3.11 always has a global minimum. Therefore, as discussed in section 3.3, the message-passing algorithms derived from the Lagrangian, equation 3.12, always possess at least one fixed point (see Yuille, 2002, for an algorithm that is guaranteed to find a minimum of F_R^K).

The following result gives sufficient conditions on R for the problem of equation 3.11 to have precisely one minimum:

Theorem 1. *The Kikuchi free energy functional, equation 3.8, is strictly convex on Δ_R^K (and hence the constrained minimization problem has a unique solution) if for all subcollections of regions $S \subseteq R$, the overcounting factors satisfy:*

$$\sum_{s \in \mathcal{F}(S)} c_s \geq 0, \tag{3.13}$$

where, as defined in section 2, $\mathcal{F}(S) := \cup_{s \in S} \mathcal{F}(s) = \{r \in R : \exists\, s \in S \text{ s.t. } r \subseteq s\}$ is the up-set of S.

Proof. See appendix A.

Corollary 1. *(cf. theorem 3 in Aji & McEliece, 2001). In the Bethe case, the constrained minimization problem of equation 3.11 has a unique solution if the graphical representation G_R of R has at most one loop.*

Proof. See Pakzad (2004).

Once we define a suitable notion of graphical representation for a general collection of regions in the next section, we will generalize the result of corollary 1.

4 Graphical Representations of the Kikuchi Approximation Problem

In this section, we define the notion of graphical representations for a Kikuchi approximation problem. The algorithms of the type discussed in section 3.3 can then be viewed as message-passing algorithms along the edges of such graphs. We will discuss this in detail in section 5.

We will further introduce minimal graphical representations for a given collection R of regions, which are graphical representations with the fewest number of edges. Our motivation for introducing such minimal graphs is twofold.

First, note that the results of section 3.4 refer to the uniqueness of the solution of the constrained minimization problem of equation 3.11. However, we are also interested in the conditions under which these solutions are the exact marginals of the product distribution of equation 3.10. As we will show in this section, the exactness of approximations obtained using equation 3.11 corresponds directly to the nonexistence of loops in the minimal graphs. In fact, we will show that in the loop-free case, this graph is a junction tree, and the message-passing algorithms of type discussed in section 3.3 correspond to a variation of junction tree algorithm.

Second, as we will discuss in detail in section 5, the message-passing algorithms of the type mentioned in section 3.3 on the minimal graphs will be the most compact among all graphical representations of the same problem, and can result in algorithms that are significantly less complex than the ones discussed in Yedidia et al. (2002) and McEliece and Yildrim (2003).

Let G be a directed acyclic graph with vertex set $V(G)$ and edge set $\mathcal{E}(G)$. Parallel to our definitions in section 2, for each vertex $r \in V(G)$, define:

$$\begin{aligned}
&\text{Ancestors:} && \mathcal{A}_G(r) := \{s \in V \,:\, \exists \text{ a directed path from } s \text{ to } r\}\\
&\text{Descendants:} && \mathcal{D}_G(r) := \{s \in V \,:\, r \in \mathcal{A}_G(s)\}\\
&\text{Parents:} && \mathcal{P}_G(r) := \{s \in V \,:\, (s \to r) \in \mathcal{E}(G)\}\\
&\text{Children:} && \mathcal{C}_G(r) := \{s \in V \,:\, (r \to s) \in \mathcal{E}(G)\}\\
&\text{Forebears:} && \mathcal{F}_G(r) := \{r\} \cup \mathcal{A}_G(r)
\end{aligned}$$

As in section 2, for a subset $S \subseteq V(G)$, we define $\mathcal{F}_G(S) := \bigcup_{s\in S} \mathcal{F}_G(s)$. Also define the depth of each vertex $r \in V(G)$ as

$$d_G(r) := \begin{cases} 0 & \text{if } \mathcal{P}_G(r) = \emptyset \\ 1 + \max_{s\in\mathcal{P}_G(r)} d_G(s) & \text{otherwise.} \end{cases}$$

Similarly we define the depth of each edge $(t \to u)$ of G as the depth of the child vertex u:

$$d_G(t \to u) := d_G(u).$$

Note that given a poset R of regions, the above definitions for the Hasse diagram G_R are consistent with the corresponding definitions for the poset from section 2 (i.e., for all $r \in V(G_R)$, $\mathcal{A}_{G_R}(r) = \mathcal{A}(r)$ and so on).

Back to the problem at hand, let R be a collection of regions as before, and let G be a directed acyclic graph whose nodes correspond to the regions $r \in R$. We will further assume that an edge $(s \to t)$ exists in G only if $t \subset s$.

Definition 3. *The edge constraint for an edge $(s \to t)$ of G is defined as the following functional of the pseudomarginals $\{b_r\ ,\ r \in R\}$:*

$$\mathrm{EC}_{(s\to t)}(\{b_r\ ,\ r \in R\}) := \sum_{\mathbf{x}_{s\setminus t}} b_s(\mathbf{x}_s) - b_t(\mathbf{x}_t). \tag{4.1}$$

When the arguments are clear from the context, we abbreviate this as $EC_{(s\to t)}$.

Definition 4. *We call G a graphical representation of Δ_R^K if Δ_R^K can be represented using the edge constraints of G, that is,*

$$\Delta_R^K = \Bigg\{ \{b_r(\mathbf{x}_r),\ r \in R\}\ :\ \forall (s \to t) \in \mathcal{E}(G),\ \mathrm{EC}_{(s\to t)} = 0$$
$$\text{and } \forall r \in R,\ \sum_{\mathbf{x}_r} b_r(\mathbf{x}_r) = 1 \Bigg\}. \tag{4.2}$$

As mentioned in the previous sections, a poset R is most naturally represented by its Hasse diagram G_R; a Hasse diagram uses the transitivity of partial ordering to represent a poset in the most compact form. Note that our local consistency constraints also have the transitivity property:

If $(r \to s)$, $(s \to t)$ and $(r \to t)$ are edges in graph G, then

$$(\mathrm{EC}_{(r\to s)} = 0) \text{ and } (\mathrm{EC}_{(s\to t)} = 0) \Longrightarrow (\mathrm{EC}_{(r\to t)} = 0).$$

Therefore, the last edge (between "grandfather" and "grandchild") is redundant. This is why the Hasse diagram G_R is a graphical representation of Δ_R^K.

On the other hand, local consistency relations satisfy a property other than transitivity, which can be used to further reduce the representation of Δ_R^K:

Suppose $(r \to s)$, $(r \to u)$, $(s \to t)$, and $(u \to t)$ are edges in graph G, then

$$(\mathrm{EC}_{(r\to s)} = 0) \text{ and } (\mathrm{EC}_{(r\to u)} = 0) \text{ and}$$
$$(\mathrm{EC}_{(u\to t)} = 0) \Longrightarrow (\mathrm{EC}_{(s\to t)} = 0).$$

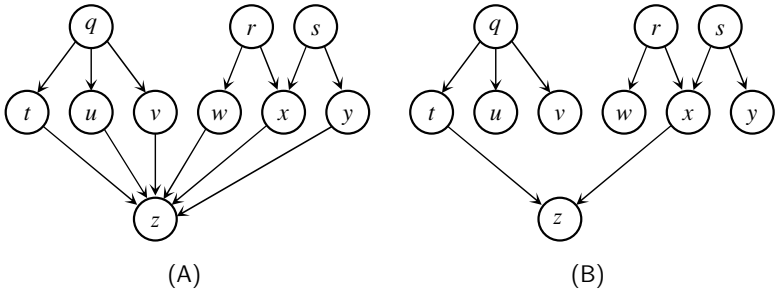

Figure 3: Equivalence of edges for removal. In the Hasse diagram, G_R depicted in *A*, $\{(t \to z), (u \to z), (v \to z)\}$ and $\{(w \to z), (x \to z), (y \to z)\}$ are the EER classes of z. (B) A realization of S_R, the minimal graphical representation of R.

Then a graph obtained by removing the edge $(s \to t)$ of G is still a graphical representation of Δ_R^K since the edge constraint of $(s \to t)$ is implied by other edge constraints. We now make precise the reductions in the graphical representation that are implied by this property.

Definition 5. *Edges* $(u \to r)$ *and* $(v \to r)$ *are said to be equivalent edges for removal (EER), and denoted* $(u \to r) \sim (v \to r)$*, if there exists a sequence* $(t_0 \to r), \ldots, (t_k \to r)$ *of edges in* G_R*, with* $t_0 = u$ *and* $t_k = v$ *and with the property that* $\forall\, i = 1, \ldots, k,\ \mathcal{A}(t_{i-1}) \cap \mathcal{A}(t_i) \neq \emptyset$.

It is easy to verify that this relation $\sim$ is reflexive, symmetrical, and transitive and is hence indeed an equivalence relation. Therefore, for each region $r \in R$, the collection of all the edges leading to r can be partitioned into equivalence classes of edges for removal (*EER classes* of region r). In the example of Figure 3A, $(t \to z) \sim (u \to z) \sim (v \to z)$ since $\mathcal{A}(t) \cap \mathcal{A}(u) = \mathcal{A}(u) \cap \mathcal{A}(v) = \{q\}$; also, $(w \to z) \sim (x \to z) \sim (y \to z)$ since $\mathcal{A}(w) \cap \mathcal{A}(x) = \{r\}$ and $\mathcal{A}(x) \cap \mathcal{A}(y) = \{s\}$. But $(v \to z) \not\sim (w \to z)$ since no sequence of edges with the desired property can be found. It follows that $\{(t \to z), (u \to z), (v \to z)\}$ and $\{(w \to z), (x \to z), (y \to z)\}$ are the EER classes of z.

Definition 6. *From each EER class, remove all but one (representative) edge from the Hasse diagram* G_R*. Denote the resulting graph by* S_R.

Figure 3B shows one realization of S_R for the Hasse diagram of Figure 3A.

Note that graph S_R is not unique, since the representative edge of each equivalence class can be arbitrarily chosen. However, the number of the edges in any choice of S_R is unique and equals the total number of EER classes of all regions. We further show in Pakzad (2004) that important

graph-theoretic properties such as the number of loops and connected components are invariant in all choices of S_R, and the subgraph of any choice of S_R on an up-set T of R is a version of S_T for T. Based on these justifications and the next proposition, we call S_R the minimal graphical representation, or the minimal graph, of R, and freely talk about the existence of loops in S_R as if S_R were unique. All results in the remainder of this article apply to every choice of S_R.

Proposition 3. *S_R is indeed a minimal graphical representation of Δ_R^K, that is, a collection of pseudomarginals $\{b_r, r \in R\}$ lies in Δ_R^K iff it satisfies all the edge constraints of S_R, and further, removal of any of the edges of S_R results in misrepresentation of Δ_R^K.*

Proof. See Pakzad (2004).

As we have seen, to solve the constrained minimization problem, one forms the Lagrangian, introducing multipliers $\lambda_{tr}(\mathbf{x}_r)$ for each edge $(t \to r)$ of S_R. Since S_R has fewer edges than any other graphical representation of R, algorithms based on S_R require the fewest message updates per each iteration.

4.1 Connection with Junction Trees. In this section we show that there is a close connection between the minimal graphical representation of a collection R of Kikuchi regions and the junction trees on R.

Definition 7. *Let $\{r_1, \ldots, r_M\}$ be a collection of subsets of the index set $[N]$. A tree or forest G with vertices $\{r_1, \ldots, r_M\}$ is called a junction tree or forest if it satisfies condition* **A**1 *of section 3.2, that is, for each $i \in [N]$, the subgraph consisting of all the vertices that contain i can be connected.*[6]

Although junction trees are traditionally defined as undirected trees, in the above definition we do not make a distinction between directed and undirected graphs; we call a directed graph a junction tree if replacing all the directed edges with undirected ones yields a junction tree in the usual sense.

Let $\{r_1, \ldots, r_M\}$ be the maximal elements of R. For the rest of this article, we assume that R is totally connected, that is, it has property **A**n for all $n = 1, \ldots, N$. The following proposition links the concept of minimal graphs to the junction trees on the maximal elements of R:

Proposition 4. *If S_R has no loops, then it is a junction forest, and hence the maximal elements $\{r_1, \ldots, r_M\}$ can be put on a junction tree. Conversely, if $\{r_1, \ldots, r_M\}$ can be put on a junction tree, then S_R has no loops.*

[6] Note that given that G has no loops, condition **A**1 implies **A**n for all n.

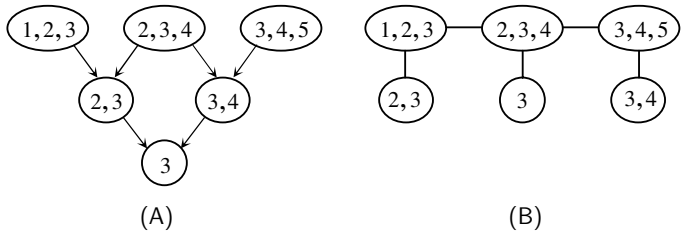

Figure 4: (A) Hasse diagram. (B) Junction tree.

Proof. See Pakzad (2004).

Recall that in section 3.2 we originally defined the Kikuchi problem in terms of a collection R_0 of regions. We now define the concept of loopiness of a Kikuchi problem:

Definition 8. *The original collection R_0 of regions is called loop free if there exists a junction tree on its maximal elements and is called loopy if no such junction tree exists.*

Now as before, let R be any poset of regions with the same maximal regions as R_0, which is totally connected. Then by proposition 4 and definition 8, R_0 is called loopy iff S_R has a loop.

4.2 Necessary and Sufficient Conditions for Exactness of the Kikuchi Method. It is well known that the belief propagation algorithm converges to the exact marginals, in finite time, if the underlying graph is loop free (see, e.g., Pearl, 1988; Cowell et al., 1999). Likewise, the message-passing algorithms of the type discussed in section 3.3 will converge to yield the exact marginals if the Hasse diagram is loop free, and in fact the value of the Kikuchi functional equals the variational free energy. However, this is a rather weak result, since only very rarely will the Hasse diagram be loop free. In fact, many collections of regions that can be put on a junction tree result in Hasse diagrams that have loops. For example the poset $R = \{\{123\}, \{234\}, \{345\}, \{23\}, \{34\}, \{3\}\}$ will have a loop in the Hasse diagram as displayed in Figure 4A, but it can be easily handled as a junction tree, as in Figure 4B. Also, in the example of Figure 3, although the Hasse diagram has loops, the solution to the Kikuchi approximation problem equals the exact marginals. This is because not all the loops of G_R are "bad" loops that cause trouble for the message-passing algorithm. In fact these "bad" loops are precisely the loops that cannot be broken when one creates S_R. In the examples of both Figures 3 and 4, S_R is loop free. One can therefore run a message-passing algorithm on S_R, which converges to yield the solution for the Kikuchi approximation, equation 3.11, which is identical to the exact

marginals. In fact, in these examples, the Kikuchi free energy functional equals the variational free energy. The results in this subsection make these observations precise.

The following theorem states sufficient conditions for the Kikuchi approximate free energy and the consistency constraint set of pseudomarginals to be exact:

Theorem 2. *(Exactness of the Kikuchi approximations, Δ_R^K and F_R^K).*

A. $\Delta_R^K = \Delta_R$ *if* S_R *is loop free.*

B. Let $b(\mathbf{x})$ *be a distribution with marginals* $b_r(\mathbf{x}_r)$*. Then* $F_R^K(\{b_r,\ r \in R\}) = F(b)$ if $b(\mathbf{x}) = \prod_{r\in R} b_r(\mathbf{x}_r)^{c_r}$.

Proof. Part A: Suppose S_R is loop free, and let $\{b_r(\mathbf{x}_r),\ r \in R\} \in \Delta_R^K$. Since S_R is a junction forest, there is a distribution $b(\mathbf{x}) := \prod_{r\in R} b_r(\mathbf{x}_r) / \prod_{(t\to u)\in\mathcal{E}(S_R)} b_u(\mathbf{x}_u)$ that marginalizes to $\{b_r(\mathbf{x}_r),\ r \in R\}$. This is a well-known result on the junction trees, which can be verified by marginalizing $b(\mathbf{x})$ in stages, from the leaves (of the undirected version of S_R) toward an arbitrary region r as the root, where at each step, by local consistency there will be cancellation. Therefore, $\{b_r(\mathbf{x}_r),\ r \in R\} \in \Delta_R$, and so $\Delta_R^K \subseteq \Delta_R$. But clearly $\Delta_R \subseteq \Delta_R^K$, since the true marginals of any distribution are locally consistent. Therefore, $\Delta_R^K = \Delta_R$.

Part B: From discussion of section 3.1, $F_R^K(\{b_r,\ r \in R\}) = F(b)$ if the entropy approximation of equation 3.5 is exact. Now

$$
\begin{aligned}
& b(\mathbf{x}) = \prod_{r\in R} b_r(\mathbf{x}_r)^{c_r} \\
\Longrightarrow\ & \mathrm{KL}\left(b(\mathbf{x}) \Big|\Big| \prod_{r\in R} b_r(\mathbf{x}_r)^{c_r}\right) = 0 \\
\Longrightarrow\ & \sum_{\mathbf{x}} b(\mathbf{x}) \left(\log(b(\mathbf{x})) - \log\left(\prod_{r\in R} b_r(\mathbf{x}_r)^{c_r}\right)\right) = 0 \\
\Longrightarrow\ & \sum_{\mathbf{x}} b(\mathbf{x}) \left(\log(b(\mathbf{x})) - \sum_{r\in R} c_r \log(b_r(\mathbf{x}_r))\right) = 0 \\
\Longrightarrow\ & \sum_{\mathbf{x}} b(\mathbf{x}) \log(b(\mathbf{x})) = \sum_{r\in R} c_r \sum_{\mathbf{x}} b(\mathbf{x}) \log(b_r(\mathbf{x}_r)) \\
\Longrightarrow\ & \sum_{\mathbf{x}} b(\mathbf{x}) \log(b(\mathbf{x})) = \sum_{r\in R} c_r \sum_{\mathbf{x}_r} b_r(\mathbf{x}_r) \log(b_r(\mathbf{x}_r)) \\
\Longrightarrow\ & F_R^K(\{b_r(\mathbf{x}_r),\ r \in R\}) = F(b(\mathbf{x})).
\end{aligned}
$$

Corollary 2. *If R_0 is loop free, then the constrained minimization problem of equation 3.11 has a unique solution. Further, the solution $\{b_r^*,\ r \in R\}$ is the exact marginal of the product function, and the minimum free energy equals the log-partition function, that is, $b_r^*(\mathbf{x}_r) = B_r(\mathbf{x}_r)$ and $F^* = -\log(Z)$.*[7]

Proof. See Pakzad (2004).

The above results show that a sufficient condition for the exactness of the solutions of the Kikuchi approximation method of equation 3.11 is that S_R be loop free. In the sequel, we address the necessary conditions for exactness.

We first pose the following more abstract question about entropy approximations: Under what conditions is an entropy approximation in the form of equation 3.5 exact for all R-decomposable distributions? The following theorem attempts to answer this question.

Theorem 3. *Let R be a totally connected poset of subsets of $[N]$, and let $\{k_r\ ,\ r \in R\}$ be a collection of constants. Then the following are equivalent:*

1. $H(B) = \sum_{r\in R} k_r\, H_r(B_r)$ *for all R-decomposable distributions $B(\mathbf{x})$.*

2. $\sum_{r\in\cup_{i\in s}\mathcal{F}(i)} k_r = 1$ *for all $s \subseteq [N]$ such that $\cup_{i\in s}\mathcal{F}(i)$ is connected in S_R.*

3. $\sum_{r\in\cup_{s\in S}\mathcal{F}(s)} k_r = 1$ *for all $S \subseteq 2^{[N]}$ such that $\cup_{s\in S}\mathcal{F}(s)$ is connected in S_R.*

Further, given the poset R, there exists a collection $\{k_r\ ,\ r \in R\}$ satisfying 1, 2, and 3 above, iff S_R, the minimal graph of R, is loop free. If such a collection $\{k_r\ ,\ r \in R\}$ exists, then it is unique and equals the (Möbius) overcounting factors $\{c_r\ ,\ r \in R\}$.

Proof. See Pakzad (2004).

An immediate consequence of this theorem is, as stated in proposition 1, that the only collection of factors for which the Kikuchi approximate free energy is exact is the (Möbius) overcounting factors, $\{c_r, r \in R\}$, as defined in section 3.1.

As mentioned in section 3.2, given a product distribution with kernels $\{\alpha_r^0(\mathbf{x}_r), r \in R_0\}$, where the regions in R_0 cannot be put on a junction tree, it is expected that expanding the collection R_0 by adding subsets of $r \in R_0$ as further regions would improve the quality of approximation obtained by the iterative algorithms such as GBP. The following result, however, shows that it is improbable that exact solutions will be obtained.

[7] In fact, when R_0 is loop free, iterative algorithms such as GBP, which we discuss in section 5, converge in finite time to the unique solutions b_r^*.

Theorem 4. *If R_0 is loopy, then except on a set of measure zero of choices of kernels $\{\alpha_r^0(\mathbf{x}_r),\ r \in R_0\}$, the Kikuchi approximation method of equations 3.11 will not produce exact results for both F_0 and $\{B_r\}$.*

Proof. Let T denote the set of R_0-kernels $\{\alpha_r^0(\mathbf{x}_r)\}$ for which the Kikuchi entropy approximation of equation 3.5 is exact for the Boltzmann distribution. Specifically, define

$$T := \left\{ \{\alpha_r^0(\mathbf{x}_r),\ r \in R_0\} : \sum_{\mathbf{x}} B(\mathbf{x}) \log(B(\mathbf{x})) \right.$$
$$\left. = \sum_{r \in R} c_r \sum_{\mathbf{x}_r} B_r(\mathbf{x}_r) \log(B_r(\mathbf{x}_r)) \right\}, \tag{4.3}$$

where, as in section 3.2, $B(\mathbf{x}) = \frac{1}{Z} \prod_{r \in R_0} \alpha_r^0(\mathbf{x}_r)$ and B_r's are its marginals. For each region $r \in R_0$, define $q_r := \prod_{i \in r} q_i$ to be the cardinality of the range of $\mathbf{x}_r$, and let $l := \sum_{r \in R_0} q_r$. Let $f : \mathbb{R}_+^l \longrightarrow \mathbb{R}$ be the error function in approximating entropy as in equation 3.5, that is,

$$\begin{aligned} f(\{\alpha_r^0(\mathbf{x}_r)\}) &:= \sum_{\mathbf{x}} B(\mathbf{x}) \log(B(\mathbf{x})) - \sum_{r \in R} c_r \sum_{\mathbf{x}_r} B_r(\mathbf{x}_r) \log(B_r(\mathbf{x}_r)) \\ &= \sum_{\mathbf{x}} \frac{\prod_{s \in R} \alpha_s^0(\mathbf{x}_s)}{\sum_{\mathbf{x}'} \prod_{s \in R} \alpha_s^0(\mathbf{x}'_s)} \log \left(\frac{\prod_{s \in R} \alpha_s^0(\mathbf{x}_s)}{\sum_{\mathbf{x}'} \prod_{s \in R} \alpha_s^0(\mathbf{x}'_s)} \right) \\ &\quad - \sum_{r \in R} c_r \sum_{\mathbf{x}_r} \frac{\sum_{\mathbf{x}_{[N]\setminus r}} \prod_{s \in R} \alpha_s^0(\mathbf{x}_s)}{\sum_{\mathbf{x}'} \prod_{s \in R} \alpha_s^0(\mathbf{x}'_s)} \\ &\quad \times \log \left(\frac{\sum_{\mathbf{x}_{[N]\setminus r}} \prod_{s \in R} \alpha_s^0(\mathbf{x}_s)}{\sum_{\mathbf{x}'} \prod_{s \in R} \alpha_s^0(\mathbf{x}'_s)} \right). \end{aligned}$$

Then $T = f^{-1}(0)$. Function $f(\cdot)$ above is clearly analytic on its domain, $\mathbb{R}_+^l$. Then, as demonstrated in Federer (1969), either $T = \mathbb{R}_+^l$ or $\mu(T) = 0$, where $\mu(\cdot)$ is the Lebesgue measure. The first alternative requires that f be identically zero on $\mathbb{R}_+^l$. But from theorem 3, it is evident that if R_0 is loopy, then the entropy approximation cannot be exact. This completes the proof.

A stronger version of this result can also be derived. In Pakzad (2004), we show that the conditional measure of the set of kernels $\{\alpha_r^0(\mathbf{x}_r), r \in R_0\}$, conditioned to be consistent with a predefined, weakly regulated pattern of zeros of the product distribution, for which the Kikuchi approximation method of equation 3.11 is exact, is zero.

We conclude this section with a generalization of corollary 1 on sufficient conditions for convexity of the Kikuchi free energy. In particular, we make a

(purely set-theoretic) connection between the number of loops of S_R and the sufficient conditions of theorem 5 for convexity of the Kikuchi free energy.

Definition 9. *We call a collection R of Kikuchi regions normal, if for all $S \subset R$, there exists a largest region (with respect to set inclusion) $m_S \in \bigcap_S \mathcal{D}(s)$, possibly the empty set, that contains any other region $u \in \bigcap_S \mathcal{D}(s)$.*

Note that the notion of normality defined here puts minor restrictions on the choices of collections R of regions, and in most cases of interest, such as the cluster variation method (Yedidia et al., 2001) and the junction graph method (Aji & McEliece, 2001), the collections of regions are in fact normal. Normality simply ensures that the minimal graph will contain no small, bow-tie-shaped loops with precisely two maximal and two minimal elements.

Theorem 5. *Let R be a normal collection of Kikuchi regions. Then the Kikuchi free energy functional $F_R^K(\{b_r\})$ is strictly convex if S_R has zero or one loop. In particular, the Kikuchi free energy for the cluster variation method of Yedidia et al. (2001) is strictly convex if S_R has zero or one loop.*

Proof. See Pakzad (2004).

5 Generalized Belief Propagation Algorithm

We are now in position to describe a class of iterative message-passing algorithms that try to solve the constrained minimization problem, equation 3.11. These algorithms can be viewed as extensions of the GBP algorithm of Yedidia et al. (2001, 2002) and the poset-BP algorithm of McEliece and Yildrim (2003). More precisely, the earlier algorithms are defined and derived only on the full Hasse diagram. The results derived in the earlier sections of this article on the minimal graphs allow us to propose algorithms for solving equation 3.11 that are often substantially less complex than the ones proposed in Yedidia et al. (2002) and McEliece and Yildrim (2003), and which appear to have comparable convergence performance in some examples we have investigated and reported in section 6.

Let R be the collection of regions for a Kikuchi approximation problem. In section 3.3 we described how the Lagrange multipliers method can be used to obtain an iterative, message-passing algorithm with fixed points that coincide with the stationary points of equation 3.11.

Now let G be any graphical representation of Δ_R^K as defined in section 4. Then the Lagrangian of equation 3.12 can be rewritten in terms of the edge constraints of G, in which case the "messages" of the resulting iterative algorithm can be identified precisely with the edges of G. This means that for each graphical representation of Δ_R^K, there is a distinct message-passing algorithm along the edges of that graph. Clearly all such algorithms have

the same set of fixed points, although the dynamics of each algorithm may be different.

So far we have represented the constraint set Δ_R^K using the edgeconstraints defined in section 4. Motivated by an observation made by Yedidia et al. (2001, 2002), we introduce an alternative but essentially equivalent set of edge constraints. We will then be able to use this alternative representation of the constraint set Δ_R^K to derive an alternative message-passing algorithm to solve equation 3.11.

Definition 10. *The YFW edge-constraint*[8] *for an edge* $(s \to t)$ *of* G *is defined as the following functional of the pseudo marginals* $\{b_r\ ,\ r \in R'\}$*:*

$$\mathrm{EC}'_{(s\to t)}(\{b_r\ ,\ r \in R'\}) := \sum_{u \in \mathcal{F}(t)\setminus\mathcal{F}(s)} c_u \sum_{\mathbf{x}_{u\setminus t}} b_u(\mathbf{x}_u), \tag{5.1}$$

where $R' := \{r \in R,\ c_r \neq 0\}$ *is the collection of regions with nonzero overcounting factors. Note that since* $c_u = 0$ *for* $u \in R'$, $\mathrm{EC}'_{(s\to t)}$ *is a function of only* $\{b_r, r \in R'\}$ *as claimed in equation 5.1. When the arguments are clear from the context, we abbreviate these edge constraints as* $\mathrm{EC}'_{(s\to t)}$.

Proposition 5. *The collection of pseudomarginals represented by the YFW edge constraints is equal to the restriction of* Δ_R^K *to* R'*. Namely, if we define*

$$\Delta'_R := \Bigg\{ \{b_r(\mathbf{x}_r),\ r \in R'\}\ :\ \forall (s \to t) \in \mathcal{E}(G),\ \mathrm{EC}'_{(s\to t)}(\{b_r, r \in R'\}) = 0$$
$$\text{and } \forall r \in R',\ \sum_{\mathbf{x}_r} b_r(\mathbf{x}_r) = 1 \Bigg\}, \tag{5.2}$$

then $\Delta'_R = \Delta_R^K|_{R'}$*, where* $\Delta_R^K|_{R'}$ *is the restriction of* Δ_R^K *to* R'*, that is, a collection* $\{b_r(\mathbf{x}_r), r \in R'\}$ *lies in* $\Delta_R^K|_{R'}$ *iff it has an extension* $\{b_r(\mathbf{x}_r), r \in R\} \in \Delta_R^K$.

Proof. See appendix B.

Remark. Note from equations 3.8 and 3.10 that the Kikuchi free energy can be rewritten as follows:

$$F_R^K(\{b_r(\mathbf{x}_r)\}) = \sum_{r\in R} \sum_{\mathbf{x}_r} (-c_r\, b_r(\mathbf{x}_r) \log(\beta_r(\mathbf{x}_r)) + c_r\, b_r(\mathbf{x}_r) \log(b_r(\mathbf{x}_r))) \tag{5.3}$$

From equation 5.3, it is apparent that $F_R^K(\{b_r, r \in R\})$ depends only on $\{b_r, r \in R'\}$, since the terms involving the pseudomarginals corresponding

[8] We call these constraints YFW after Yedidia, Freeman, and Weiss.

to the regions with zero overcounting factors are multiplied by zero. Therefore, equation 3.11 can be rewritten as

$$\min_{\{b_r, r\in R\}\in\Delta_R^K} F_R^K(\{b_r, r\in R\}) = \min_{\{b_r, r\in R'\}\in\Delta'_R} F_R^K(\{b_r, r\in R'\})$$

$$\text{and } \left(\arg\min_{\{b_r, r\in R\}\in\Delta_R^K} F_R^K(\{b_r, r\in R\})\right)\Bigg|_{R'} = \arg\min_{\{b_r, r\in R'\}\in\Delta'_R} F_R^K(\{b_r, r\in R'\}). \tag{5.4}$$

In other words, the central constrained minimization problem of equation 3.11 is reduced to the following: $\min_{\{b_r, r\in R'\}\in\Delta'_R} F_R^K(\{b_r, r\in R'\})$. This observation is also made in Yedidia et al. (2002).

We will now write the Lagrangian for equation 5.4 using the YFW edge constraints:

$$\mathcal{L} := \sum_{r\in R} c_r\, b_r(\mathbf{x}_r)\log\left(\frac{b_r(\mathbf{x}_r)}{\beta_r(\mathbf{x}_r)}\right) + \sum_{(r\to t)\in\mathcal{E}(G)}\sum_{\mathbf{x}_t}\lambda_{rt}(\mathbf{x}_t)$$
$$\times \sum_{u\in\mathcal{F}(t)\setminus\mathcal{F}(r)} c_u \sum_{\mathbf{x}_{u\setminus t}} b_u(\mathbf{x}_u) + \sum_{r\in R}\kappa_r\left(\sum_{\mathbf{x}_r} b_r(\mathbf{x}_r) - 1\right). \tag{5.5}$$

Setting partial derivative $\partial\mathcal{L}/\partial b_r(\mathbf{x}_r) = 0$ for each region r and each value of $\mathbf{x}_r$, and identifying messages, for each edge $(p\to r)$, as $m_{pr}(\mathbf{x}_r) := e^{-\lambda_{pr}(\mathbf{x}_r)}$, we obtain:

$$b_r(\mathbf{x}_r) = k\,\beta_r(\mathbf{x}_r)\left(\prod_{p\in\mathcal{P}_G(r)} m_{pr}(\mathbf{x}_r)\right)\left(\prod_{d\in\mathcal{D}(r)}\prod_{p'\in\mathcal{P}_G(d)\setminus(\{r\}\cup\mathcal{D}(r))} m_{p'd}(\mathbf{x}_d)\right), \tag{5.6}$$

where constant k is chosen to normalize b_r so it will sum to 1, and message m_{pr} is updated to satisfy the original edge constraint $\sum_{\mathbf{x}_{p\setminus r}} b_p(\mathbf{x}_p) - b_r(\mathbf{x}_r) = 0$:

$$m_{pr}(\mathbf{x}_r) =$$
$$k'\frac{\sum_{\mathbf{x}_{p\setminus r}}\beta_p(\mathbf{x}_p)\left(\prod_{s\in\mathcal{P}_G(p)} m_{sp}(\mathbf{x}_p)\right)\left(\prod_{d\in\mathcal{D}(p)}\prod_{s'\in\mathcal{P}_G(d)\setminus(\{p\}\cup\mathcal{D}(p))} m_{s'd}(\mathbf{x}_d)\right)}{\beta_r(\mathbf{x}_r)\left(\prod_{s\in\mathcal{P}_G(r)\setminus\{p\}} m_{sr}(\mathbf{x}_r)\right)\left(\prod_{d\in\mathcal{D}(r)}\prod_{p'\in\mathcal{P}_G(d)\setminus(\{r\}\cup\mathcal{D}(r))} m_{p'd}(\mathbf{x}_d)\right)}, \tag{5.7}$$

where k' is any convenient constant. Note that the common terms from the numerator and denominator of equation 5.7 can be cancelled, but to avoid even more complex expressions, we will not write the explicit form here.

The fixed points of equations 5.6 and 5.7 set all the derivatives of the Lagrangian equal to zero, and hence are precisely the stationary points of the Kikuchi free energy F_R^K subject to constraint set Δ_R^K.

The algorithm of equations 5.6 and 5.7 is defined on any graphical representation of Δ_R^K and has as many messages as the edges of the underlying graph. From the results of section 4, using S_R, the minimal graphical representation, yields the least complex such algorithm in this sense. In fact, in most cases, the algorithm on S_R is substantially less complex than the full version implemented on the Hasse diagram G_R.

Remark. Note that the GBP algorithm presented here is not fundamentally different from its original form as presented in Yedidia et al. (2001, 2002); in fact, McEliece and Yildrim (2003) described an algorithm called Poset-BP, which is equivalent to the restriction of our results when G is the Hasse diagram. However, it is important to note that our results show that in general, there are iterative algorithms with strictly fewer messages, and potentially simpler dynamics, that have the same fixed points. In particular, the messages corresponding to the edges of the Hasse diagram that are removed in forming a more compact graphical representation can be set to 1 in the entire algorithm. In addition, the update rules for the remaining messages are also less complex, since they depend on fewer edges.

It is also worth mentioning that the proofs of the correctness for the GBP and Poset-BP algorithms given in Yedidia et al. (2002) and McEliece and Yildrim (2003) both presume that the poset is first simplified by removing the regions with zero overcounting factors. We note, however, that removing the regions with zero overcounting factors can in general alter the problem. This is because a region with zero overcounting factor may still serve to ensure consistency between the pseudomarginals at other regions (see, for example, the poset in Figure 5). We have avoided this restriction by proving the results for a general poset.

Consider now the restriction of the above algorithm in the Bethe case, that is, each region in R is either maximal or minimal with regard to inclusion. Then $\mathcal{P}(r) = \emptyset$ for a maximal region r and $\mathcal{D}(s) = \emptyset$ for a minimal region s. To demonstrate the connection with the belief propagation algorithm, in

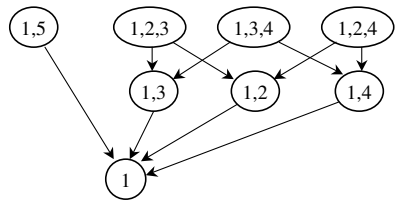

Figure 5: The overcounting factor for region 1 is zero, but it cannot be removed.

addition to the messages $m_{pr}(\mathbf{x}_r)$ for $r \subset p$, we also define messages from a child to parent as follows:

$$n_{rp}(\mathbf{x}_r) := \beta_r(\mathbf{x}_r) \prod_{s \in \mathcal{P}(r) \setminus \{p\}} m_{sr}(\mathbf{x}_r). \tag{5.8}$$

Then, by equation 5.6, for a maximal region $p \in R$,

$$b_p(\mathbf{x}_p) = k\, \beta_p(\mathbf{x}_p) \prod_{d \in \mathcal{D}(p)} n_{pd}(\mathbf{x}_d). \tag{5.9}$$

Similarly, for a minimal region $r \in R$,

$$b_r(\mathbf{x}_r) = k\, \beta_r(\mathbf{x}_r) \prod_{d \in \mathcal{P}(r)} m_{dr}(\mathbf{x}_r). \tag{5.10}$$

The update equation 5.7 for messages m_{pr} can then be rewritten as

$$\begin{aligned} m_{pr}(\mathbf{x}_r) &= k' \frac{\sum_{\mathbf{x}_{p \setminus r}} \beta_p(\mathbf{x}_p) \prod_{d \in \mathcal{D}(p)} n_{dp}(\mathbf{x}_d)}{\beta_r(\mathbf{x}_r)\, n_{rp}(\mathbf{x}_r)} \\ &= k' \sum_{\mathbf{x}_{p \setminus r}} \frac{\beta_p(\mathbf{x}_p)}{\beta_r(\mathbf{x}_r)} \prod_{d \in \mathcal{D}(p) \setminus \{r\}} n_{dp}(\mathbf{x}_d). \end{aligned} \tag{5.11}$$

It is now easy to see that equations 5.8 to 5.11 precisely define the conventional belief propagation algorithm of Pearl (1988) applied on G_R.

6 Experimental Results

In the previous section, we proved that the fixed points of GBP algorithms on any graphical representation for a poset R coincide with the solutions to the Kikuchi approximation problem of section 3.2. We further argued that the algorithm on the minimal graph S_R has the smallest complexity per each iteration. Two important questions are not addressed in this article so far: How close are the Kikuchi approximations to the true marginals? How does the convergence behavior of the GBP algorithm on the minimal graph S_R compare to that on the full Hasse graph G_R? In this section, we address these questions with some experiments.

We considered three simple loopy posets below. In each case, all the variables were binary. For each run of the experiment for a given poset, first we generated a random collection of kernels $\{\alpha_r(\mathbf{x}_r)\}$, where each value $\alpha_r(\mathbf{x}_r)$ was chosen independently and uniformly in the interval $[0, 1]$. Next we calculated the product distribution $B(\mathbf{x}) = \prod_{r \in R} \alpha_r(\mathbf{x}_r)$ together with its true

marginals $B_r(\mathbf{x}_r)$. The GBP algorithm of section 5 then was run on each of the two graphs G_R and S_R for that poset. Further, two different schedules were incorporated to update the messages for each algorithm: parallel and serial. With the parallel schedule, all messages were updated together at each iteration. For the serial schedule, we update the messages one after another, in an order chosen so as to minimize the number of edges that are updated before their requisite set of edges has been updated. Each message is updated exactly once during each iteration. To ensure convergence of some algorithms, we used damping in the update rule for the messages. The quantity w reported for each algorithm is the damping factor. In particular, we used $m_{pr}^{n+1}(\mathbf{x}_r) = w\, F(\{m^n\}) + (1-w)\, m_{pr}^n(\mathbf{x}_r)$, where m_{pr}^n is the message at iteration n and $F(\{m^n\})$ is the pure update rule of equation 5.7. The value of w is always between 0 and 1, with $w = 1$ corresponding to equation 6.7. For each case, we decreased w gradually to ensure that the algorithm converged.

For each poset, we report the savings in complexity per each iteration of GBP on the minimal graph compared to that on the Hasse graph. To compute these savings, we calculated the total arithmetic complexity, that is, the number of additions, multiplications, and divisions involved in update rules of equation 5.7 for both algorithms. Note that this is not simply the fraction of edges that are removed in forming the minimal graph, since the update rules for the messages that remain in the minimal graph are less complex than the ones on the Hasse graph.

To summarize the performance of each algorithm, at each iteration we calculated a special measure of distance between the beliefs $\{b_r\}$ and the true marginals $\{B_r\}$. We define a distance function $D(b_r, B_r) := \frac{\max_{\mathbf{x}_r} |b_r(\mathbf{x}_r) - B_r(\mathbf{x}_r)|}{\max_{\mathbf{x}_r} B_r(\mathbf{x}_r)}$ as the measure of distance from the belief b_r to the marginal B_r; this is a normalized maximum point-wise difference between the two distributions. The closer D is to 0, the closer the belief $b_r(\mathbf{x}_r)$ is to the true marginal $B_r(\mathbf{x}_r)$ at all configurations of $\mathbf{x}_r$. At each iteration we then calculate the maximum distance $\max_{r\in R} D(b_r, B_r)$, and the mean distance $\frac{1}{|R|}\sum_{r\in R} D(b_r, B_r)$. For each poset, the averages of these quantities over 200 runs are reported.

Poset 1. The Hasse diagram of this poset has one loop, but the minimal graph is loop free (see Figure 6). There is a saving of 35.7% per each iteration of GBP on the minimal graph compared to that on the Hasse graph.

As expected, the Kikuchi approximations coincide with the true marginals in this loop-free case. The serial algorithms converge to the fixed points after one iteration, because we use an optimal schedule for activating the messages. The parallel algorithm on the minimal graph takes four iterations (equal to the girth of the graph). The parallel algorithm on the Hasse graph requires damping and converges much more slowly. Note that in this case, the algorithm on the minimal graph both gives better performance at each iteration and has less complexity per iteration (see Figure 7).

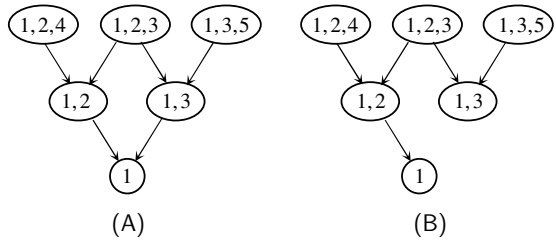

Figure 6: Poset 1. (A) Hasse graph G_R. (B) Minimal graph S_R.

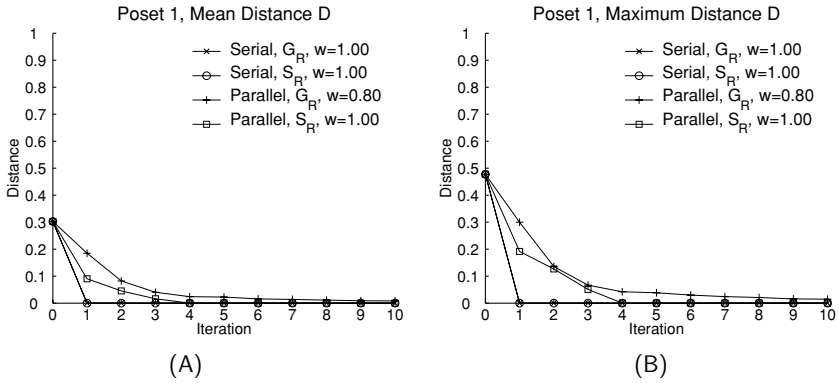

Figure 7: GBP performance on Poset 1. (A) Mean distance. (B) Maximum distance.

Poset 2. The Hasse diagram of this poset has five loops; all but one of these loops are broken in the minimal graph (see Figure 8). There is a saving of 46.2% per each iteration of GBP on the minimal graph compared to that on the Hasse graph. The Kikuchi approximations are at an average distance of about 0.05 from the true marginals, while the worst estimates have distance of about 0.13. Again, the serial algorithms converge very quickly, although the one on the Hasse graph requires a slight damping. Comparing the parallel algorithms, the one on the minimal graph clearly outperforms the one on the full Hasse graph, even with equal damping factors (see Figure 9).

Poset 3. The Hasse diagram of this poset has five loops, whereas the minimal graph has only two loops (see Figure 10). There is a saving of 28.5% per each iteration of GBP on the minimal graph compared to that on the Hasse graph. The Kikuchi approximations are at an average distance of about 0.05 from the true marginals, while the worst estimates have distance of about 0.14. Once again, the serial algorithms converge very quickly, without the need for damping. The parallel algorithm on the

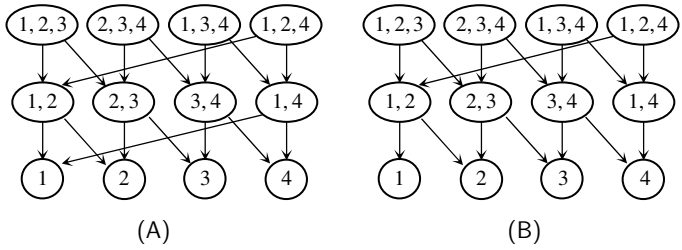

Figure 8: Poset 2. (A) Hasse graph G_R. (B) Minimal graph S_R.

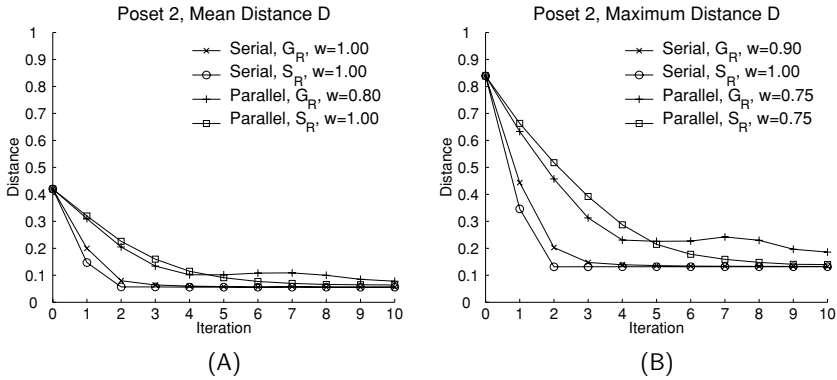

Figure 9: GBP performance on Poset 2. (A) Mean distance. (B) Maximum distance.

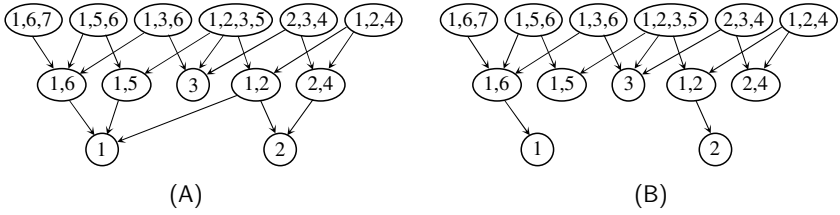

Figure 10: Poset 3. (A) Hasse graph G_R. (B) Minimal graph S_R.

minimal graph again outperforms that on the full Hasse graph, the latter requiring a damping factor $w = 0.70$ to avoid oscillations (see Figure 11).

At least for the simple posets considered here, the less complex GBP algorithm on the minimal graph, developed in this article, seems to perform better than the full GBP on the Hasse graph, especially with the parallel versions of the algorithm. Considering that each iteration of the algorithm on the minimal graph is less complex than that on the full Hasse graph, this

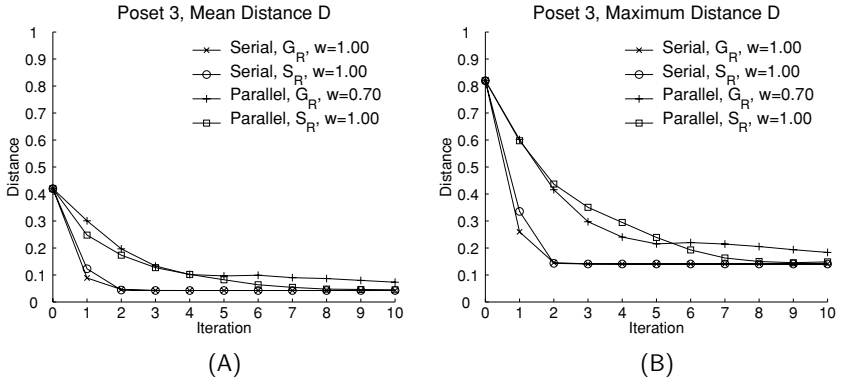

Figure 11: GBP performance on Poset 3. (A) Mean distance. (B) Maximum distance.

suggests that there is considerable saving in the complexity to be gained by using the algorithm on the minimal graph.

Appendix A: Proof of Theorem 1

Note that the Kikuchi approximate free energy, as a functional of the pseudomarginals $\{b_r(\mathbf{x}_r)\} \in \Delta_R^K$, consists of an energy term, which is linear, and a linear combination of entropy terms, with both positive and negative coefficients. We will show that if the hypothesis of the theorem holds, there is a matching between the negative and the positive terms such that the overall entropy term will be a positive linear combination of Kullback-Leibler (KL) divergence terms that is strictly convex (see, e.g., Cover & Thomas, 1991). We will prove the existence of such matching using results from the bipartite graph theory.

Form a bipartite graph $G(V^+, V^-, E)$ with vertex sets V^+ and V^- and the edge set E as follows:

- For each $r \in R$ with $c_r < 0$, create $|c_r|$ nodes $\{v_r^1, \cdots, v_r^{|c_r|}\}$ in V^-.
- For each $s \in R$ with $c_s > 0$, create c_s nodes $\{u_s^1, \cdots, u_s^{c_s}\}$ in V^+.
- To form the edge set E, connect each $v_i^r \in V^-$ to each $u_j^s \in V^+$ iff $r \subset s$.

For a subset $S \subseteq V^-$, denote by $N(S)$ the subset of nodes in V^+ that are connected to a node in S. Then graph G has the following property:

$$\forall S \subseteq V^-, \quad |S| \leq |N(S)|. \tag{A.1}$$

To see this, let $S = \{v_s^i : (s,i) \in \mathcal{I}\}$ where the index set $\mathcal{I}$ consists of some pairs of the form (s,i) with $c_s < 0$ and $0 < i \leq |c_s|$. Now create an-

other index set $\overline{\mathcal{I}}$ as $\overline{\mathcal{I}} := \{(s,\ j)\ :\ (s,i) \in \mathcal{I}\ \text{ for some } i\ ,\ 0 < j \le |c_s|\}$, and let $\overline{S} := \{v_s^i\ :\ (s,i) \in \overline{\mathcal{I}}\}$. Then clearly $S \subseteq \overline{S}$ and hence $|S| \le |\overline{S}|$, but notice that $N(S) = N(\overline{S})$. Also note that $|\overline{S}| = -\sum_{t\in T} c_t$, where $T := \{t \in R\ :\ (t,1) \in \overline{\mathcal{I}}\}$. Further,

$$\begin{aligned}\sum_{t\in\mathcal{A}(T)} c_t &= \sum_{t\in\mathcal{A}(T);c_t>0} c_t + \sum_{t\in\mathcal{A}(T);c_t<0} c_t \\ &= |N(\overline{S})| + \sum_{t\in\mathcal{A}(T);c_t<0} c_t \\ &\le |N(\overline{S})|,\end{aligned}$$

where the second equality follows from the definitions of $|N(\overline{S})|$ and $\mathcal{A}(T)$. But by the hypothesis of the theorem, $-\sum_{t\in T} c_t \le \sum_{t\in\mathcal{A}(T)} c_t$. Putting these together, we get $|S| \le |\overline{S}| = -\sum_{t\in T} c_t \le \sum_{t\in\mathcal{A}(T)} c_t \le |N(\overline{S})| = |N(S)|$ as claimed.

Then the bipartite graph satisfies the hypothesis of Hall's matching theorem (Hall, 1935), and hence there is a matching on G that saturates every vertex of V^-. In other words, there is matching $M = \{(v_r^i, u_s^j)\}$ such that every $v_r^i \in V^-$ is uniquely matched with a $u_s^j \in V^+$. Denote by U the subset of vertices in V^+ that are left unmatched.

We now rewrite the entropy term of the Kikuchi free energy, that is, the second summation in equation 3.8, using the matching M. For each $\{b_r\} \in \Delta_R^K$:

$$\begin{aligned}&\sum_{r\in R} c_r \sum_{\mathbf{x}_r} b_r \log(b_r) \\ &= \sum_{r:c_r<0} c_r \sum_{\mathbf{x}_r} b_r \log(b_r) + \sum_{s:c_s>0} c_s \sum_{\mathbf{x}_s} b_s \log(b_s) \\ &= -\sum_{r:c_r<0} \sum_{i=1}^{-c_r} \sum_{\mathbf{x}_r} b_r \log(b_r) + \sum_{s:c_s>0} \sum_{j=1}^{c_s} \sum_{\mathbf{x}_s} b_s \log(b_s) \\ &= -\sum_{v_r^i\in V^-} \sum_{\mathbf{x}_r} b_r \log(b_r) + \sum_{u_s^j\in V^+} \sum_{\mathbf{x}_s} b_s \log(b_s) \\ &= \sum_{(v_r^i,u_s^j)\in M} \left(\sum_{\mathbf{x}_s} b_s \log(b_s) - \sum_{\mathbf{x}_r} b_r \log(b_r) \right) + \sum_{u_s^j\in U} \sum_{\mathbf{x}_s} b_s \log(b_s) \\ &= \sum_{(v_r^i,u_s^j)\in M} \sum_{\mathbf{x}_s} b_s \log\left(\frac{b_s}{b_r}\right) + \sum_{u_s^j\in U} \sum_{\mathbf{x}_s} b_s \log(b_s). \qquad \text{(A.2)}\end{aligned}$$

Notice that for each $(v_r^i, u_s^j) \in M$, by definition of the bipartite graph G, we have $r \subset s$. Further, we have taken $\{b_r\} \in \Delta_R^K$, and so that $\sum_{\mathbf{x}_{s\setminus r}} b_s(\mathbf{x}_s) = b_r(\mathbf{x}_r)$, which implies the last equality.

Now note that the first term in equation A.2 is a sum of KL divergences,[9] which are strictly convex as functions of their arguments, and the second term is a sum of negative entropy functions which are also strictly convex (see, e.g., Cover & Thomas, 1991). On the other hand, as mentioned earlier, the average energy term of the Kikuchi free energy—the first summation in equation 3.8—is linear in $\{b_r\}$. Since, constrained by Δ_R^K, the Kikuchi free energy is in effect a functional only of the pseudomarginals associated with the maximal regions in R, and since each maximal region contributes such a KL divergence term in equation A.2, the Kikuchi functional as a whole is also strictly convex.

Appendix B: Proof of Proposition 5

Given $t \in R, s \in \mathcal{P}_G(t)$, by definition of the overcounting factors,

$$\sum_{u\in\mathcal{F}(t)} c_u = 1 \quad \text{and} \quad \sum_{u\in\mathcal{F}(s)} c_u = 1$$

$$\text{Therefore,} \quad \sum_{u\in\mathcal{F}(t)\setminus\mathcal{F}(s)} c_u = 0. \tag{B.1}$$

Now if $\{b_r,\ r \in R\} \in \Delta_R^K$, then $\forall\, u \in \mathcal{F}(t),\ \sum_{\mathbf{x}_{u\setminus t}} b_u(\mathbf{x}_u) = b_t(\mathbf{x}_t)$. Therefore,

$$\begin{aligned}
\mathrm{EC}'_{(s\to t)}(\{b_r\ ,\ r \in R'\}) &= \sum_{u\in\mathcal{F}(t)\setminus\mathcal{F}(s)} c_u \sum_{\mathbf{x}_{u\setminus t}} b_u(\mathbf{x}_u) \\
&= \sum_{u\in\mathcal{F}(t)\setminus\mathcal{F}(s)} c_u\, b_t(\mathbf{x}_t) \\
&= b_t(\mathbf{x}_t) \sum_{u\in\mathcal{F}(t)\setminus\mathcal{F}(s)} c_u \\
&= 0.
\end{aligned}$$

Hence, $\{b_r,\ r \in R'\} \in \Delta'_R$, and we have proven that $\Delta_R^K|_{R'} \subseteq \Delta'_R$.

Now, conversely, suppose that $\{b_r,\ r \in R'\} \in \Delta'_R$. We will show by induction on depth function $d(t)$ of region $t \in R$ (with regard to poset R, and not graph G) that for all $s \in \mathcal{A}(t)$, $\sum_{\mathbf{x}_{s\setminus t}} b_s(\mathbf{x}_s) = b_t(\mathbf{x}_t)$. The statement holds for the maximal regions, since these regions cannot have parents. Now let

[9] To be precise, each term differs from a true KL divergence by a constant.

t be a region with depth $d(t) = l > 0$, and let $\mathcal{P}_G(t) = \{s_1, \ldots, s_m\}$. For each pair s_i and s_j of parents of t in G, consider the following cases on $\mathcal{A}(s_i) \cap \mathcal{A}(s_j)$:

- Suppose $\mathcal{A}(s_i) \cap \mathcal{A}(s_j) = \emptyset$. Then, because $\{b_r, r \in R'\} \in \Delta'_R$, we have

$$\sum_{u \in \mathcal{F}(t) \setminus \mathcal{F}(s_i)} c_u \sum_{\mathbf{x}_{u \setminus t}} b_u(\mathbf{x}_u) = 0$$

$$\sum_{u \in \mathcal{F}(t) \setminus \mathcal{F}(s_j)} c_u \sum_{\mathbf{x}_{u \setminus t}} b_u(\mathbf{x}_u) = 0.$$

  Subtracting one from another, we obtain the following equality:

$$\sum_{u \in \mathcal{F}(s_i)} c_u \sum_{\mathbf{x}_{u \setminus t}} b_u(\mathbf{x}_u) = \sum_{u \in \mathcal{F}(s_j)} c_u \sum_{\mathbf{x}_{u \setminus t}} b_u(\mathbf{x}_u). \tag{B.2}$$

  Since $d(s_i)$ and $d(s_j)$ are each no larger than $l - 1$, by induction hypothesis, we have

$$\forall\, u \in \mathcal{F}(s_i), \quad \sum_{\mathbf{x}_{u \setminus s_i}} b_u(\mathbf{x}_u) = b_{s_i}(\mathbf{x}_{s_i})$$

$$\forall\, u \in \mathcal{F}(s_j), \quad \sum_{\mathbf{x}_{u \setminus s_j}} b_u(\mathbf{x}_u) = b_{s_j}(\mathbf{x}_{s_j}).$$

  Replacing these in equation B.2, we obtain

$$\sum_{\mathbf{x}_{s_i \setminus t}} b_{s_i}(\mathbf{x}_{s_i}) \sum_{u \in \mathcal{F}(s_i)} c_u = \sum_{\mathbf{x}_{s_j \setminus t}} b_{s_j}(\mathbf{x}_{s_j}) \sum_{u \in \mathcal{F}(s_j)} c_u.$$

  But by definition of the overcounting factors, $\sum_{u \in \mathcal{F}(s_i)} c_u = \sum_{u \in \mathcal{F}(s_j)} c_u = 1$, so that $\sum_{\mathbf{x}_{s_i \setminus t}} b_{s_i}(\mathbf{x}_{s_i}) = \sum_{\mathbf{x}_{s_j \setminus t}} b_{s_j}(\mathbf{x}_{s_j})$.

- Suppose $u \in \mathcal{A}(s_i) \cap \mathcal{A}(s_j)$. Then again by induction hypothesis, we have $\sum_{\mathbf{x}_{u \setminus s_i}} b_u(\mathbf{x}_u) = b_{s_i}(\mathbf{x}_{s_i})$ and $\sum_{\mathbf{x}_{u \setminus s_j}} b_u(\mathbf{x}_u) = b_{s_j}(\mathbf{x}_{s_j})$. Thus, once again,

$$\sum_{\mathbf{x}_{s_i \setminus t}} b_{s_i}(\mathbf{x}_{s_i}) = \sum_{\mathbf{x}_{u \setminus t}} b_u(\mathbf{x}_u) = \sum_{\mathbf{x}_{s_j \setminus t}} b_{s_j}(\mathbf{x}_{s_j}).$$

We can therefore show that for all pairs s_i and s_j of parents of t in G, $\sum_{\mathbf{x}_{s_i \setminus t}} b_{s_i}(\mathbf{x}_{s_i}) = \sum_{\mathbf{x}_{s_j \setminus t}} b_{s_j}(\mathbf{x}_{s_j}) = b'_t(\mathbf{x}_t)$ for a unique function $b'_t(\mathbf{x}_t)$. Now if $t \notin R'$, we define $b_t(\mathbf{x}_t) := b'_t(\mathbf{x}_t)$. If $t \in R'$, using the fact that $\{b_r,\ r \in R'\} \in \Delta'_R$, we have

$$\sum_{u \in \mathcal{F}(t) \setminus \mathcal{F}(s_i)} c_u \sum_{\mathbf{x}_{u \setminus t}} b_u(\mathbf{x}_u) = 0$$

$$\implies \quad c_t\, b_t(\mathbf{x}_t) + \sum_{u \in \mathcal{A}(t) \setminus \mathcal{F}(s_i)} c_u\, b'_t(\mathbf{x}_t) = 0$$

$$\implies \quad b_t(\mathbf{x}_t) = b'_t(\mathbf{x}_t),$$

since by equation B.1, $c_t + \sum_{u \in \mathcal{A}(t) \setminus \mathcal{F}(s_i)} c_u = 0$, and $c_t \neq 0$.

We have shown that $\sum_{\mathbf{x}_{s_i \setminus t}} b_{s_i}(\mathbf{x}_{s_i}) = b_t(\mathbf{x}_t)$ for all $s_i \in \mathcal{P}_G(t)$. But G is a graphical representation of Δ_R^K. Therefore, by argument similar to those of proposition 3 for each $(s \to t) \in \mathcal{E}(G_R) \setminus \mathcal{E}(G)$, the edge constraint $\sum_{\mathbf{x}_{s \setminus t}} b_s(\mathbf{x}_s) = b_t(\mathbf{x}_t)$ is implied by the edge constraints of those edges of G at the same, or at a lower, depth. Specifically, there must be a path in G between u and t for each $u \in \mathcal{A}(t)$, consisting only of vertices that contain t, or else consistency between b_u and b_t could not be implied by the edge constraints of G. But any vertex that contains t must have a depth less than t (remember that we are using the depth function on R, and not on G: a region containing t could have a G depth higher than that of t.) Therefore, all the G edges in this path have depths no more than $l = d(t)$ and can be used in our inductive argument. Together, they imply the consistency between u and t, that is, $\sum_{\mathbf{x}_{u \setminus t}} b_u(\mathbf{x}_u) = b_t(\mathbf{x}_t)$. Therefore, we have found the desired extension $\{b_r,\ r \in R\} \in \Delta_R$, and so $\Delta'_R \subseteq \Delta_R^K\big|_{R'}$. This proves that $\Delta'_R = \Delta_R^K\big|_{R'}$ as claimed.

Acknowledgments

This work was supported by grants from ONR/MURI, N00014-1-0637; NSF, SBR-9873086; DARPA, F30602-00-2-0538; California Micro Program; Texas Instruments; Marvell Technologies; and ST MicroElectronics.

References

Aji, S., & McEliece, R. (2000). The generalized distributive law. *IEEE Trans. Inform. Theory*, *46*(2), 325–343.

Aji, S., & McEliece, R. (2001). The generalized distributive law and free energy minimization. In *Proceedings of the Allerton Conference on Communication, Control, and Computing* (pp. 672–681). Urbana, IL: University of Illinois.

Berrou, C., Glavieux, A., & Thitimajshima, P. (1993). Near Shannon limit error-correcting coding and decoding: Turbo-codes. In *Proceedings of the IEEE International Conference on Communications* (no. 2, pp. 1064–1070). Piscataway, NJ: IEEE.

Cover, T., & Thomas, J. (1991). *Elements of information theory*. New York: Wiley.

Cowell, R., Dawid, A., Lauritzen, S., & Spiegelhalter, D. (1999). *Probabilistic networks and expert systems*. Berlin: Springer-Verlag.

Divsalar, D., Jin, H., & McEliece, R. (1998). Coding theorems for "turbo-like" codes. In *Proceedings of the Allerton Conference on Communication, Control, and Computing* (pp. 201–210). Urbana, IL: University of Illinois.

Federer, H. (1969). *Geometric measure theory*. Berlin: Springer-Verlag.

Gallager, R. (1963). *Low-density parity-check codes*. Cambridge, MA: MIT Press.

Hall, P. (1935). On representatives of subsets. *Journal of London Mathematical Society*, *10*, 26–30.

Kikuchi, R. (1951). A theory of cooperative phenomena. *Phys. Rev.*, *6*(81), 988–1003.

Kittel, C., & Kroemer, H. (1980). *Thermal physics*. New York: W. H. Freeman.

Luby, M. (2002). LT-codes. In *Proceedings of IEEE Symposium on the Foundations of Computer Science* (pp. 271–280). Piscataway, NJ: IEEE.

MacKay, D., & Neal, R. (1995). Good codes based on very sparse matrices. In *Cryptography and Coding: 5th IMA Conference* (pp. 100–111). Berlin: Springer-Verlag.

McEliece, R., MacKay, D., & Cheng, J. (1998). Turbo decoding as an instance of Pearl's "belief propagation" algorithm. *IEEE J. Select. Areas Commun.*, *16*(2), 140–152.

McEliece, R., & Yildrim, M. (2003). Belief propagation on partially ordered sets. In J. Rosenthal & D. S. Gilliam (Eds.), *Mathematical systems theory in biology, communications, computation, and finance* (pp. 275–300). Berlin: Springer.

Morita, T. (1994). Formal structure of the cluster variation method. *Prog. Theor. Phys. Supp.*, *115*, 27–39.

Pakzad, P. (2004). *Low complexity, high performance algorithms for estimation and decoding*. Unpublished doctoral dissertation, University of California, Berkeley.

Pakzad, P., & Anantharam, V. (2002a). Belief propagation and statistical physics. In *Conference on Information Sciences and Systems (CISS 2002)* (Paper No. 225). Princeton, NJ: Princeton University.

Pakzad, P., & Anantharam, V. (2002b). Minimal graphical representations of Kikuchi regions. In *Proceedings of the Allerton Conference on Communication, Control, and Computing*. Piscataway, NJ: IEEE.

Pakzad, P., & Anantharam, V. (2004). A new look at the generalized distributive law. *IEEE Trans. Inform. Theory*, *50*(6), 1132–1155.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. San Mateo, CA: Morgan Kaufmann.

Richardson, T. (2000). The geometry of turbo-decoding dynamics. *IEEE Trans. Inform. Theory*, *46*(1), 9–23.

Richardson, T., Shokrollahi, A., & Urbanke, R. (2001). Design of capacity-approaching irregular low-density parity-check codes. *IEEE Trans. Inform. Theory*, *47*(2), 619–637.

Richardson, T., & Urbanke, R. (2001). The capacity of low-density parity-check codes under message-passing decoding. *IEEE Trans. Inform. Theory*, *47*(2), 599–618.

Stanley, R. (1986). *Enumerative combinatorics* (Vol. 1). Monterey, CA: Wadsworth & Brooks/Cole.

Tanner, R. (1981). A recursive approach to low complexity codes. *IEEE Trans. Inform. Theory*, *27*(9), 533–547.

Wainwright, M., & Jordan, M. (2003). *Graphical models, exponential families, and variational inference* (Tech. Rep. 649). Berkeley: Department of Statistics, University of California.

Walrand, J., & Varaiya, P. (1996). *High-performance communication networks*. San Francisco: Morgan Kaufmann.

Weiss, Y. (2000). Correctness of local probability propagation in graphical models with loops. *Neural Computation*, *12*, 1–41.

Welling, M., & Teh, Y. (2001). Belief optimization for binary networks: A stable alternative to loopy belief propagation. In *Proceedings of the International Conference on Uncertainty in Artificial Intelligence* (pp. 554–561). San Francisco: Morgan Kaufmann.

Wicker, S. (1995). *Error control systems for digital communication and storage*. Upper Saddle River, NJ: Prentice Hall.

Yedidia, J., Freeman, W., & Weiss, Y. (2001). *Bethe free energy, Kikuchi approximations, and belief propagation algorithms* (Tech. Rep. TR2001-16). Cambridge, MA: Mitsubishi Electric Research Lab.

Yedidia, J., Freeman, W., & Weiss, Y. (2002). *Constructing free energy approximations and generalized belief propagation algorithms* (Tech. Rep. TR2002-35). Cambridge, MA: Mitsubishi Electric Research Lab.

Yuille, A. (2002). CCCP algorithms to minimize the Bethe and Kikuchi free energies: Convergent alternatives to belief propagation. *Neural Computation*, *14*, 1691–1722.

Received March 19, 2004; accepted January 31, 2005.