ESTIMATION AND MARGINALIZATION USING KIKUCHI APPROXIMATION METHODS

by

Payam Pakzad and Venkat Anantharam

Memorandum No. UCB/ERL M04/21

14 December 2003

ESTIMATION AND MARGINALIZATION USING KIKUCHI APPROXIMATION METHODS

by

Payam Pakzad and Venkat Anantharam

Memorandum No. UCB/ERL M04/21

1 January 2003

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

Estimation and Marginalization using Kikuchi Approximation Methods

Payam Pakzad (payamp@eecs.berkeley.edu)
Venkat Anantharam (ananth@eecs.berkeley.edu)

12/14/2003
Berkeley, California

Abstract

In this paper we examine a general method of approximation, – known as *Kikuchi approximation method*, – for finding the marginals of a product-distribution. 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. In this paper 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. We give conditions under which such algorithms converge to a stationary point of the optimization problem. Implementation of these algorithms on graphs with fewer edges require fewer operations in each iteration. We therefore characterize *minimal graphs* for a Kikuchi problem, which are those with the minimum number of edges, and show that all such minimal graphs have the same number of loops and share several important connectivity properties. We show that if the minimal graph is cycle-free, then Kikuchi approximation method is exact and the converse is also true generically; together with the fact that in the cycle-free case the above-mentioned iterative algorithms are equivalent to the well-known belief propagation algorithm, our results imply that, generically, 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-function is encountered frequently in various branches of science and engineering. An important special case, the *probabilistic inference* problem (Cowell et al., 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 which is 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 codeword $\mathbf{x} := (x_1, \cdots, x_n)$, with noisy observations $\mathbf{y}^* := (y_1^*, \cdots, y_n^*)$ 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 codewords, as well as conditional probabilities representing the noisy observations:

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

where $1(\cdot)$ 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 mainly interested 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 and Varaiya, 1996), that the invariant distribution has the form

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

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 $R_j \subset \{1, \cdots, M\}$ is the index set of routes that use link j. Finally Z is the partition function, defined by

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

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

As another example, in thermal 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 and 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 and McEliece, 2000; Morita, 1994; Yedidia et al., 2001; Luby, 2002; Pakzad and 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 and Neal, 1995) and turbo codes (Berrou et al., 1993) which are decoded using instances of the belief propagation algorithm on a loopy graph (McEliece et al., 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 and Urbanke, 2001; Weiss, 2000; Richardson et al., 2001; Divsalar et al., 1998; Richardson, 2000; MacKay and Neal, 1995).

It was shown recently in (Yedidia et al., 2001) 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 paper, the fixed points of the belief propagation algorithm were shown to coincide with the stationary points of *Bethe free energy* subject to consistency constraints. Here, 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 *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 Kikuchi free energy. Using such message-passing algorithms is expected to result in approximations that are closer to the marginals than are the ones given by belief propagation.

In this paper we will 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 et al., 2001), (Yedidia et al., 2002) and (McEliece and Yildrim, 2003). Furthermore, we will show that, for generic problems, 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.

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

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

The outline of this paper 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 which approximates the desired marginals as the constrained fixed points of an appropriately-defined 'free energy functional', and further 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 results to the Bethe case gives a strengthening of the famous single-loop criterion of

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 Kikuchi approximation. In Section 5 we derive the generalized belief propagation (GBP) algorithm of (Yedidia et al., 2001) on any arbitrary graphical representation of a Kikuchi problem. This is a generalization of results in (Yedidia et al., 2002) and (McEliece and Yildrim, 2003). Some experimental results are reported in Section 6, comparing the convergence properties of GBP algorithm as presented in (Yedidia et al., 2002) and (McEliece and Yildrim, 2003), with the most compact GBP algorithm derived in this paper.

2 Problem Setup

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

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

Here Z is the normalizing constant and is called 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 paper 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 paper 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 ($\forall r \in R, r \subseteq r$), antisymmetric ($r \subseteq s$ 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. 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, i.e. an edge $(t \rightarrow u)$ exists in G_R iff $u \prec t$. □

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 paper 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:

Ancestors:	$\mathcal{A}(r) := \{s \in R \ : \ r \subset s\}$
Descendants	$\mathcal{D}(r) := \{s \in R \ : \ s \subset r\}$
Forebears (Up-set)	$\mathcal{F}(r) := \{s \in R \ : \ r \subseteq s\}$

Further for $r \in R$ we define

Parents	$\mathcal{P}(r) := \{s \in R \ : \ r \prec s\}$
Children	$\mathcal{C}(r) := \{s \in R \ : \ s \prec r\}$

Note that in each of these definitions, the collection of subsets being defined is comprised of regions, even though the argument r of $\mathcal{A}(r), \mathcal{D}(r)$ and $\mathcal{F}(r)$ need

[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 paper.

not be a region itself. For a collection S of subsets of $[N]$, we define $\mathcal{F}(S) := \bigcup_{s \in S} \mathcal{F}(s)$. 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, i.e. $E(\mathbf{x}) = \sum_{r \in R} E_r(\mathbf{x}_r)$ for certain functions $\{E_r(\mathbf{x}_r), r \in R\}$.

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

$$F(b(\mathbf{x})) := U(b(\mathbf{x})) - H(b(\mathbf{x})) \tag{2}$$

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

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

where $B(\mathbf{x})$ is the Boltzmann distribution of (1). Then the variational free energy can be rewritten as follows:

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

where $\mathrm{KL}(b||B)$ is the Kullback-Leibler divergence between $b(\mathbf{x})$ and $B(\mathbf{x})$, see e.g. (Cover and Thomas, 1991). It is then clear that $F(b)$ is uniquely minimized when $b(\mathbf{x})$ equals the Boltzmann distribution $B(\mathbf{x})$ of (1), and we have

$$F_0 := \min_{b(\mathbf{x})} F(b(\mathbf{x})) = F(B(\mathbf{x})) = -\log(Z). \tag{3}$$

As mentioned in the introduction, equation (3) 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), 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) 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. 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-pseudo-marginals.*
A collection of R-pseudo-marginals that are further 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 $\mathbf{x}$, i.e. 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 (3) as

$$\begin{aligned} F_0 &= \min_{\{b_r(\mathbf{x}_r)\} \in \Delta_R} F_R(\{b_r(\mathbf{x}_r)\}) \\ \{b_r^*(\mathbf{x}_r)\} &= \arg \min_{\{b_r(\mathbf{x}_r)\} \in \Delta_R} F_R(\{b_r(\mathbf{x}_r)\}) \end{aligned} \tag{4}$$

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

$$U(b(\mathbf{x})) = \sum_{r \in R} \sum_{\mathbf{x}_r} b_r(\mathbf{x}_r) E_r(\mathbf{x}_r) \tag{5}$$

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 (2) 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

$$H(b(\mathbf{x})) \simeq \sum_{r \in R} k_r\, H_r(b_r(\mathbf{x}_r)) \tag{6}$$

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,), 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:

$$c_r = 1 - \sum_{s \in \mathcal{A}(r)} c_s \tag{7}$$

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 (6), as we show in Proposition 1:

Proposition 1. *The only choice of factors $\{k_r\}$ which can result in exactness of* (6) *for all R-decomposable Boltzmann distributions ,- i.e. 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 Möbius overcounting factors $\{c_r\}$.*

We will prove this proposition in Section 4. 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 'loop-free' 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 (4)):

$$\{B_r(\mathbf{x}_r)\} \simeq \{b_r^*(\mathbf{x}_r)\} := \arg \min_{\{b_r(\mathbf{x}_r)\}\in\Delta_R^K} F_R^K(\{b_r(\mathbf{x}_r)\}) \qquad (8)$$

Here $F_R^K(\{b_r\})$, known as the *Kikuchi free energy*, see e.g. (Kikuchi, 1951), is defined as (cf. equation (35) in (Yedidia et al., 2001))

$$F_R^K(\{b_r(\mathbf{x}_r)\}) := \sum_{r\in R}\sum_{\mathbf{x}_r} b_r(\mathbf{x}_r)E_r(\mathbf{x}_r) + \sum_{r\in R}\sum_{\mathbf{x}_r} c_r\, b_r(\mathbf{x}_r)\log(b_r(\mathbf{x}_r)) \qquad (9)$$

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

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

$$\text{and } \forall u \in R,\ \sum_{\mathbf{x}_u} b_u(\mathbf{x}_u) = 1\Big\} \qquad (10)$$

Note that in general the constraints of Δ_R^K are not enough to guarantee that every collection of pseudo-marginals $\{b_r,\ r \in R\} \in \Delta_R^K$ 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 (10) 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 $b(\mathbf{x})$, i.e. $\{b_r, r \in R\}$, 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[3] 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 (1) takes the following product forms:

$$B(\mathbf{x}) = \frac{\prod_{r'\in R_0}\alpha_{r'}^0(\mathbf{x}_{r'})}{Z} = \frac{\prod_{r\in R}\alpha_r(\mathbf{x}_r)}{Z} = \frac{\prod_{r\in R}\beta_r(\mathbf{x}_r)^{c_r}}{Z} \tag{11}$$

where the last equality follows from the fact that, by (7), $\sum_{r\in\mathcal{F}(s)} c_r = 1$ for all $s \in R$.

Using approximations (9) and (10) we are now interested in solving the following:

Problem (Kikuchi Approximation):

$$\begin{aligned} -\log(Z) \simeq F^* &:= \min_{\{b_r(\mathbf{x}_r)\}\in\Delta_R^K} F_R^K(\{b_r(\mathbf{x}_r)\}) \\ \text{and}\quad \{B_r(\mathbf{x}_r)\} \simeq \{b_r^*(\mathbf{x}_r)\} &:= \arg\min_{\{b_r(\mathbf{x}_r)\}\in\Delta_R^K} F_R^K(\{b_r(\mathbf{x}_r)\}) \end{aligned} \tag{12}$$

□

Note now that by equation (4), 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 (12) would correspond exactly to

[3] Note however that the way this assignment is done can impact the quality of the approximations to (4) provided by (12).

the collection of the marginals of the product function $B(\mathbf{x})$ of equation (11); 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 (12) are expected to be close approximations to these marginals.

Our focus in the rest of this paper 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 b_r^*'s are equal to the marginals B_r of the Boltzmann distribution. We also address in detail in Section 4 message-passing algorithms on graphs which solve equation (12) which are more efficient than the ones known to date, such as the *generalized belief propagation* (Yedidia et al., 2002) and *poset belief propagation* (McEliece and Yildrim, 2003).

The collection R of regions effectively specifies both the Kikuchi approximation (9), and the constraint set (10). It is also evident that (12) 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 (12), which are parameterized by the poset[4] 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 paper we only consider 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 (6) 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 –

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

should improve the approximation (at the expense of increasing the complexity of the underlying minimization).

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

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

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

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

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 Kikuchi problem (12), 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 $(\mathbf{A1})$. 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 (6) are those in which each variable x_i appears the same number of times on the two sides of the equality sign, i.e.

$$\sum_{r \in \mathcal{F}(\{i\})} c_r = 1 \qquad \text{for each } \ i = 0, \cdots, N-1 \qquad \textbf{(B1)}$$

We can extend this condition also, as follows:

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

Conditions $(\mathbf{B}n)$ are called the *balance condtions*, 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 variational 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 non-empty intersection of regions. The following proposition shows that the choice of R made in the cluster variational method is expected to give a reasonable Kikuchi approximation.

Proposition 2. *Any collection of regions R which is closed under non-empty intersection of regions is totally connected and totally balanced.*

Proof. Note first that if $u, v \in R$ and $u \subset v$, then there is a directed path from v to u in G_R, where all the nodes in the path contain u. Let $t \neq \emptyset$ be any subset of $[N]$, and let $r, s \in \mathcal{F}(t)$ be any two regions containing t. Then $r \cap s$ must lie in R, since R is closed under non-empty intersections. Therefore r and s each are connected in G_R to $r \cap s$, where all the vertices on the paths from r to $r \cap s$ and from s to $r \cap s$ contain $r \cap s$, which in turn contains t. This proves that R is totally connected.

Now let $r \subseteq [N]$ be a subset such that $\mathcal{F}(r) \neq \emptyset$. If $r \in R$, then by definition of the overcounting factors $\sum_{t \in \mathcal{F}(r)} c_t = 1$ and we are done. Suppose then that $r \notin R$. We will show that there is a unique minimal $s \in \mathcal{F}(r)$, so that $\mathcal{F}(r) = \mathcal{F}(s)$. If not, then there must be at least two minimal regions t_1 and t_2 in $\mathcal{F}(r)$, with $t_1 \not\subseteq t_2$ and $t_2 \not\subseteq t_1$. Then $t_1 \cap t_2$ is a region, strictly smaller than both t_1 and t_2, which lies in $\mathcal{F}(r)$ since it contains r. This would contradict t_1 and t_2 each being minimal in $\mathcal{F}(r)$. Therefore there exists an $s \in R$ such that $\mathcal{F}(s) = \mathcal{F}(r)$, and therefore $\sum_{t \in \mathcal{F}(r)} c_t = \sum_{t \in \mathcal{F}(s)} c_t = 1$. This proves that R is totally balanced. □

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 paper. In this case G_R can be thought of as a hypergraph in which the maximal regions of R are the vertices and the minimal regions are the hyperedges. If we insist, as assumed in (Yedidia et al., 2001), that the maximal regions be pairs $\{i, j\}$ of indices for $i, j \in [N]$, and that the minimal regions be all the singletons $\{i\}$ for $i \in [N]$, then we will in fact have a poset R of depth 2 which is closed under intersection; this is what was called the Bethe case in (Yedidia et al., 2001). Our notion of *Bethe case* is more general than that of (Yedidia et al., 2001), since no restriction on the size of the regions is necessary, and we allow for R not to be closed under intersection.

On the other hand, (Aji and McEliece, 2001) considers only the case when the aforementioned 'hypergraph' view of G_R is a graph, i.e. 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 he 'junction graph' condition given in (Aji and McEliece, 2001) is simply the intersection of conditions (**A**1) and (**B**1) above. It can also be shown that the 'junction graph' condition of (Aji and McEliece, 2001) does *not* imply either (**A**2) or (**B**2).

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 $\{1\}, \{2\}, \cdots, \{16\}$ respectively, and the region associated with each 'check-node' is the subset of $\{1, \cdots, 16\}$ 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).

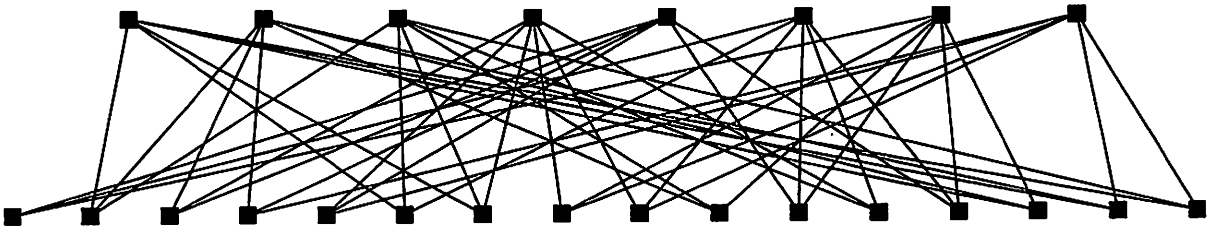


Figure 1: Tanner Graph of a Linear Code

This poset has property (**A**1), but not (**A**2): note for example that 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 (**B1**), but not (**B2**): for $s := \{2, 6\}$, $\mathcal{F}(s)$ is precisely the first and third check-node regions. Then $\sum_{r \in \mathcal{F}(s)} c_r = 1 + 1 = 2 \neq 1$.

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.

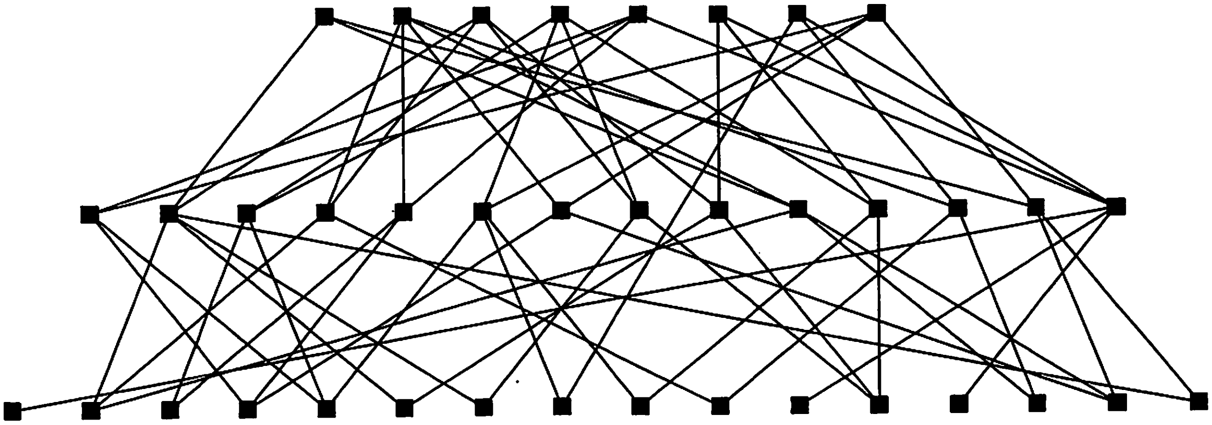


Figure 2: Alternative Poset of Linear Code of Example 3

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; e.g. 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.

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 (12). We form the Lagrangian:

$$\begin{aligned}\mathcal{L} := \sum_{r \in R} \sum_{\mathrm{x}_r} (-b_r(\mathrm{x}_r) \log(\alpha_r(\mathrm{x}_r)) + c_r\, b_r(\mathrm{x}_r) \log(b_r(\mathrm{x}_r))) \\ + \sum_{r \in R} \sum_{t \prec r} \sum_{\mathrm{x}_t} \lambda_{rt}(\mathrm{x}_t) (b_t(\mathrm{x}_t) - \sum_{\mathrm{x}_{r \setminus t}} b_r(\mathrm{x}_r)) \\ + \sum_{r \in R} \kappa_r (\sum_{\mathrm{x}_r} b_r(\mathrm{x}_r) - 1) \quad (13)\end{aligned}$$

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 $t \prec r$, i.e. 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 detailed derivation for a nice such algorithm 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. It should be noted that the algorithm we derive in Section 5, although called GBP, can be considerably less complex than the one called GBP in (Yedidia et al., 2001). This is because of certain systematic complexity-reducing transformations we carry out in Section 4, which constitute the major practical contribution of this work.

3.4 Convexity Conditions

In this section we describe our results regarding the convexity of the optimization problem (12), which we first reported in (Pakzad and Anantharam,).

Kikuchi free energy (9) constrained on $\{b_r\} \in \Delta_R^K$ is bounded below and hence the constrained minimization problem (12) always has a global minimum. Therefore, as discussed in Section 3.3, the message passing algorithms derived from Lagrangian (13) 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 (12) to have precisely one minimum:

Theorem 3. *Kikuchi free energy functional* (9) *is strictly convex on* Δ_R^K *(and* *hence the constrained minimization problem has a unique solution) if the over-counting factors $c_r, r \in R$ satisfy:*

$$\forall S \subseteq R, \quad \sum_{s \in \mathcal{F}(S)} c_s \geq 0 \tag{14}$$

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 set of forebears of* S.

Proof. Note that Kikuchi approximate free energy, as a functional of the pseudo-marginals $\{b_r(\mathbf{x}_r)\} \in \Delta_R^K$ 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 KL divergence terms which are strictly convex, see e.g. (Cover and 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:

$$\forall S \subseteq V^-, \quad |S| \leq |N(S)| \tag{15}$$

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 another index set $\overline{\mathcal{I}}$ as $\overline{\mathcal{I}} := \{(s,j) \ : \ (s,i) \in \mathcal{I} \text{ for some } i\ ,\ 0 < j \leq |c_s|\}$, and let $\overline{S} := \{v_s^i \ : \ (s,i) \in \overline{\mathcal{I}}\}$. Then clearly $S \subseteq \overline{S}$ and hence $|S| \leq |\overline{S}|$, 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,

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

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

Then the bipartite graph satisfies the hypothesis of Hall's Matching Theorem (see (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, i.e. the second summation in (9), using the matching M. For each $\{b_r\} \in \Delta_R^K$:

$$\begin{aligned}\sum_{r\in R} c_r \sum_{\mathbf{x}_r} b_r\log(b_r) &= \sum_{r:c_r<0} c_r \sum_{\mathbf{x}_r} b_r\log(b_r) + \sum_{s:c_s>0} c_s \sum_{\mathbf{x}_s} b_s\log(b_s)\\ &= -\sum_{r:c_r<0}\sum_{i=1}^{-c_r}\sum_{\mathbf{x}_r} b_r\log(b_r) + \sum_{s:c_s>0}\sum_{j=1}^{c_s}\sum_{\mathbf{x}_s} b_s\log(b_s)\\ &= -\sum_{v_r^i\in V^-}\sum_{\mathbf{x}_r} b_r\log(b_r) + \sum_{u_s^j\in V^+}\sum_{\mathbf{x}_s} b_s\log(b_s)\\ &= \sum_{(v_r^i,u_s^j)\in M}\Big(\sum_{\mathbf{x}_s} b_s\log(b_s) - \sum_{\mathbf{x}_r} b_r\log(b_r)\Big) + \sum_{u_s^j\in U}\sum_{\mathbf{x}_s} b_s\log(b_s)\\ &= \sum_{(v_r^i,u_s^j)\in M}\sum_{\mathbf{x}_s} b_s\log(\frac{b_s}{b_r}) + \sum_{u_s^j\in U}\sum_{\mathbf{x}_s} b_s\log(b_s) \qquad (16)\end{aligned}$$

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 (16) is a sum of KL-divergences[5], which are strictly convex as functions of their arguments, and the second term is a sum

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

of negative entropy functions which are also strictly convex, see e.g. (Cover and Thomas, 1991). On the other hand, as mentioned earlier, the average energy term of the Kikuchi free energy, i.e. the first summation in (9), is linear in $\{b_r\}$. Since, constrained by Δ_R^K, the Kikuchi free energy is in effect a functional only of the pseudo-marginals associated to the *maximal* regions in R, and since each maximal region contributes such a KL-divergence term in (16), the Kikuchi functional as a whole is also strictly convex. □

Corollary 4. *(cf. Theorem 3 in (Aji and McEliece, 2001)) In the Bethe case, the constrained minimization problem* (12) *has a unique solution if the graphical representation G_R of R has at most one loop.*

Proof. Let $S \subseteq R$ be a subset of regions, and consider the sum $A_S := \sum_{r \in \mathcal{F}(S)} c_r$. Note that in the Bethe case, $c_r = 1 - (\#\text{ of parents of } r)$ for all $r \in R$. This means that for each region $r \in R$, the contribution of c_r to the sum A_S can be broken up as a contribution of $(+1)$ for the vertex r, and a contribution of (-1) for each edge of the Hasse graph ending in r. Therefore, A_S is precisely equal to the number of vertices minus the number of edges of the Hasse graph of $\mathcal{F}(S)$, which is a subgraph of G_R. Therefore the sum is nonnegative iff this subgraph has at most one loop.

By Theorem 3, a sufficient condition for uniqueness of solution to the constrained minimization problem (12) is that the sum A_S above be nonnegative for all subsets $S \subseteq R$. In particular, choosing $S = R$ implies, by above, that G_R has at most one loop. On the other hand, if G_R has at most one loop, then any of its subgraphs will have no more than one loop, and by the above argument A_S will be nonnegative for each $S \subseteq R$. The consequence is that the nonnegativity of A_S for all $S \subseteq R$ is equivalent to the statement that G_R have at most one loop. Therefore the sufficient condition (14) for the uniqueness of solution is that G_R have no more than one loop. □

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 4.

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 introduction of such minimal graphs is two-fold.

First, note that the results of Section 3.4 refer to the uniqueness of solution of the constrained minimization problem (12). However, one is further interested in the conditions under which these solutions are the exact marginals of the product distribution (11). As we will show in this section, the exactness of approximations obtained using (12) corresponds directly to non-existence 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 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 such algorithms as the GBP of (Yedidia et al., 2002) and the poset-BP of (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:

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

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

$$\mathrm{EC}_{(s \to t)}(\{b_r\,, r \in R\}) := \sum_{\mathbf{x}_{s \setminus t}} b_s(\mathbf{x}_s) - b_t(\mathbf{x}_t) \tag{17}$$

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

Definition. We call G a *graphical representation* of Δ_R^K if Δ_R^K can be represented using the edge-constraints of G, i.e.

$$\Delta_R^K = \Big\{ \{b_r(\mathbf{x}_r), r \in R\} \;:\; \forall (s \to t) \in \mathcal{E}(G),\ \mathrm{EC}_{(s \to t)} = 0 \\ \text{and } \forall r \in R,\ \sum_{\mathbf{x}_r} b_r(\mathbf{x}_r) = 1 \Big\} \tag{18}$$

□

As mentioned in the previous sections, a poset R is most naturally represented by its Hasse diagram G_R; 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

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

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

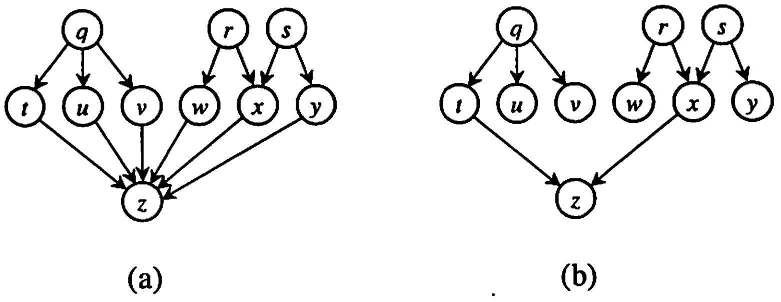
$$\left(\mathrm{EC}_{(r\rightarrow s)} = 0\right) \text{ and } \left(\mathrm{EC}_{(r\rightarrow u)} = 0\right) \text{ and } \left(\mathrm{EC}_{(u\rightarrow t)} = 0\right) \implies \left(\mathrm{EC}_{(s\rightarrow t)} = 0\right)$$

Then a graph obtained by removing the edge $(s \rightarrow t)$ of G is still graphical representation of Δ_R^K since the edge-constraint of $(s \rightarrow t)$ is implied by other edge-constraints. We will refer to this property as the $(\Diamond)$ property.

We now make precise the reductions in the graphical representation which are implied by the anti-transitivity property.

Definition. Edges $(u \rightarrow r)$ and $(v \rightarrow r)$ are said to be *Equivalent Edges for Removal* (*EER*), and denoted $(u \rightarrow r) \sim (v \rightarrow r)$ if there exists a sequence $(t_0 \rightarrow r), \cdots, (t_k \rightarrow r)$ of edges in G_R, with $t_0 = u$ and $t_k = v$ and with the property that $\forall i = 1, \cdots, k$, $\mathcal{A}(t_{i-1}) \cap \mathcal{A}(t_i) \neq \emptyset$, i.e. $\exists\, w_i \in R$ s.t. $t_{i-1} \subset w_i$ and $t_i \subset w_i$. □

Then it is easy to verify that this relation '$\sim$' is reflexive, symmetric 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 3(a), $\{t, u, v\}$ and $\{w, x, y\}$ are the EER classes of z.



(a) (b)

Figure 3: Equivalence of Edges for Removal. In the Hasse diagram (a) of R the EER classes of z are $\{t, u, v\}$ and $\{w, x, y\}$. (b) is a realization of S_R, the minimal graphical representation of R.

Definition. From each EER class $\{(t_1 \rightarrow r), \cdots, (t_m \rightarrow r)\}$, remove all but one (representative) edge from the Hasse diagram G_R. Denote the resulting graph by S_R. □

Figure 3(b) shows one realization of S_R for the Hasse diagram of part (a).

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 of any choice of S_R is unique and equals the total number of EER classes of all regions. As we will see shortly, all choices of S_R result in equivalent, minimal graphical representations of R. *All results in the remainder of this paper apply to every choice of S_R.*

Lemma 5. *For every pair $u, s \in R$ such that $s \subset u$, there is a path in S_R between s and u consisting only of nodes that contain s.*

Proof. Clearly the Hasse diagram G_R has the claimed property; in fact if $s \subset u$, then there is a *directed* path from u to s consisting only of nodes that contain s.

Also note that if the claim is true for all pairs $s \prec u$ with $d(s) \leq l$, then it is also true for all pairs $s \subset u'$ with $d(s) \leq l$; this is because $s \subset u'$ implies that there exists a sequence of regions $s = u_0 \prec u_1 \prec \cdots \prec u_k = u'$, with $d(u_i) \leq l$ for $i = 0, \cdots k-1$. Then there is a path in S_R between each pair u_i and u_{i+1} consisting only of nodes that contain u_i and hence s.

Now to prove the claim for pairs $s \prec u$ we proceed by induction. Suppose first that $s \prec u$ and $d(s) = 1$. Then the edge $(u \rightarrow s)$ of G_R remains in S_R, since edges of depth 1 cannot be EER with other edges. Next assume inductively that we have proven the claim for all pairs $s' \prec u'$ with $d(s') \leq l$, and suppose $s \prec u$ is a pair of regions of R with $d(s) = l + 1$. Then from the definition of S_R there remains an edge $(v \rightarrow s)$ in S_R, a subset $\{t_0 = u, t_1, \cdots, t_k = v\}$ of parents of s and a sequence $\{w_1, \cdots, w_k\}$ of regions of R, s.t. $w_i \in \mathcal{A}(t_{i-1}) \cap \mathcal{A}(t_i)$. Each t_i has a depth of at most l, so for each $i = 1, \cdots, k$ there are paths in S_R from w_i to t_{i-1} and to t_i consisting only of nodes that contain t_{i-1} and t_i respectively. But $s \subset t_i$ for all i, and hence there is a path $(s, v, \cdots, w_k, \cdots, t_{k-1}, \cdots, w_{k-1}, \cdots, \cdots, w_1, \cdots, u)$ in S_R consisting only of nodes that contain s. This completes the proof. □

Now suppose S_R^1 and S_R^2 are two instances of S_R. As mentioned earlier, the number of edges of S_R^1 and S_R^2 are the same. Also by Lemma 5, the connected components of any S_R correspond one-to-one to those in G_R. Therefore S_R^1 is loop-free iff S_R^2 is loop-free, and in fact the number of loops of S_R^1 is equal to that of S_R^2. With this justification and based on the next proposition, we call S_R *the minimal graphical representation*, or *the minimal graph,* of R, and freely talk about existence of loops in S_R, as if S_R were unique.

Lemma 6. *Let $T \subset R$, and view $\mathcal{F}(T)$ as a sub-poset of R. Let S_R be a minimal graphical representation of R, and let G denote the subgraph of S_R on $\mathcal{F}(T)$. Then G is a minimal graphical representation of $\mathcal{F}(T)$. Furthermore, for each $t \in \mathcal{F}(T)$, the overcounting factors of t w.r.t. posets R and $\mathcal{F}(T)$ are the same.*

Proof. For each $t \in \mathcal{F}(T)$, the EER classes of t in poset $\mathcal{F}(T)$ are identical to those in R. Hence G by definition has one edge from each EER class, making it a minimal graphical representation of $\mathcal{F}(T)$, as claimed. Similarly, the overcounting factor c_t w.r.t. R depends only on the regions in $\mathcal{F}(t)$. These are all included in $\mathcal{F}(T)$. Therefore a simple inductive argument shows that the overcounting factors of t w.r.t. R and $\mathcal{F}(T)$ are identical. □

Based on this result, in the rest of this paper, when talking about a specific choice of S_R, we write $S_{\mathcal{F}(T)}$ to denote the subgraph of S_R on $\mathcal{F}(T)$, as the choice of the minimal graphical representation of $\mathcal{F}(T)$.

Proposition 7. *S_R is indeed a 'minimal' graphical representation of Δ_R^K, i.e. a collection of pseudo-marginals $\{b_r,\, r \in R\}$ 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. To show that the collection of edge-constraints of S_R is a sufficient representation of Δ_R^K, note that by Lemma 5, for each $r \subset t$ there is a path between r and t with only nodes that contain r. Then the collection of edge-constraints of this path imply that $\sum_{\mathrm{x}_{t\setminus r}} b_t(\mathrm{x}_t) = b_r(\mathrm{x}_r)$. Therefore any collection $\{b_r,\, r \in R\}$ of pseudo-marginals satisfying all the edge-constraints of S_R belongs to Δ_R^K.

To prove minimality, let S_R' be a graph created from S_R by removing an edge $(t_1 \to u)$ of S_R. Let $\{(t_1 \to u), \cdots, (t_k \to u)\}$ be the corresponding EER class – all of these edges are now removed in creating S_R'. Define $T := \bigcup_{i=1}^k \bigcup_{r \in \mathcal{F}(t_i)} r$.

Let $b(\mathbf{x}) \geq \epsilon > 0$ be a positive distribution, with marginals $b_r(\mathbf{x}_r), r \subseteq [N]$. Then $\{b_r(\mathbf{x}_r), r \in R\} \in \Delta_R^K$. Now define another collection $\{b'_r(\mathbf{x}_r)\}$ as follows:

$$b'_T(\mathbf{x}_T) := \begin{cases} b_T(\mathbf{x}_T) + \epsilon \cdot (-1)^{\sum_{i \in u} x_i} & \text{if } x_i \in \{0,1\}\ \forall i \in u \\ b_T(\mathbf{x}_T) & \text{else} \end{cases}$$

Extend this to define $b'_r(\mathbf{x}_r)$ for all the regions $r \in R$ as follows:

$$\begin{aligned} b'_r(\mathbf{x}_r) &:= \sum_{\mathbf{x}_{T \setminus r}} b'_T(\mathbf{x}_T) && \text{if } r \in \bigcup_{i=1}^{k} \mathcal{F}(t_i) \\ b'_r(\mathbf{x}_r) &:= b_r(\mathbf{x}_r) && \text{else} \end{aligned}$$

We claim that (1) $b'_r = b_r$ for all $r \in R \setminus (\bigcup_{i=1}^k \mathcal{F}(t_i))$; (2) $b'_r \neq b_r$ for any $r \in (\bigcup_{i=1}^k \mathcal{F}(t_i))$, and (3) all the edge-constraints of S'_R are satisfied.

Part (1) is obvious from the definition of b' above.

Part (2) can be seen easily since for any $r \in \bigcup_{i=1}^k \mathcal{F}(t_i)$, $u \subset r$ and by definition, marginalizations of b'_T are different from those of b_T on the subsets that contain u.

To see part (3), let $(r_1 \rightarrow r_2)$ be any edge of S'_R. If both $r_1, r_2 \in R \setminus (\bigcup_{i=1}^k \mathcal{F}(t_i))$ or both $r_1, r_2 \in \bigcup_{i=1}^k \mathcal{F}(t_i)$, then by definition, b'_{r_1} is consistent with b'_{r_2}. If $r_2 \in \bigcup_{i=1}^k \mathcal{F}(t_i)$, then its parent, r_1 must also lie there. Therefore we need only check the case when r_1 lies in $(\bigcup_{i=1}^k \mathcal{F}(t_i))$ and r_2 does not. Note that in this case, r_2 cannot contain u.Also from definition of $b'_T(\mathbf{x}_T)$, the marginals of b'_T at the level of any subset that does not contain u coincide with the marginals of b_T since the ϵ terms cancel out. Hence b'_{r_1} marginalizes to b_{r_2} and since $b'_{r_2} = b_{r_2}$, the edge-constraint in S'_R corresponding to $(r_1 \rightarrow r_2)$ is also satisfied in this case.

We have therefore shown explicitly that there exist collections $\{b'_r\}$ of pseudo-marginals that satisfy all the edge-constraints of S'_R but not the edge-constraint corresponding to edge $(t_1 \rightarrow u)$ of S_R (in particular, $\sum_{\mathbf{x}_{t_1 \setminus u}} b'_{t_1}(\mathbf{x}_{t_1}) \neq b'_u(\mathbf{x}_u) = b_u(\mathbf{x}_u)$.) Therefore $\{b'_r\} \notin \Delta_R^K$. □

As we have seen, to solve the constrained minimization problem one forms the Lagrangian, introducing multipliers $\lambda_{tr}(\mathbf{x}_r)$ for each edge $(t \rightarrow r)$ 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. Let $\{r_1, \cdots, r_M\}$ be a collection of subsets of the index set $[N]$. A tree/forest G with vertices $\{r_1, \cdots, r_M\}$ is called a *junction* tree/forest if it satisfies condition (**A**1) of Section 3.2, i.e. that for each $i \in [N]$, the subgraph consisting of all the vertices that contain i be connected.[6] □

Although junction trees are traditionally defined as undirected trees, in the above definition we do not make 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 $\{r_1, \cdots, r_M\}$ be the maximal elements of R. For the rest of this paper we assume that R is totally connected, i.e. it has property (**A**n) for all $n = 1, \cdots, N$. Then it is easy to see from the definition above that the following proposition holds:

Proposition 8. *If S_R has no loops, then it is a junction forest and hence $\{r_1, \cdots, r_M\}$ can be put on a junction tree.*

Proof. The Hasse graph G_R satisfies (**A**1). Let $(u_0, u_1, \cdots, u_n)$ form a path in G_R where $i \in u_j$ for all $j = 0, \cdots, n$. Then for all $j = 1, \cdots, n$ either $u_{j-1} \prec u_j$ or $u_j \prec u_{j-1}$. Then by Lemma 5 there is a path between u_{j-1} and u_j for all $j = 1, \cdots, n$ consisting only of nodes that contain i. This proves that S_R satisfies (**A**1). Therefore S_R is a junction forest on R.

Now starting with undirected version of S_R, successively absorb each node $r \in R \backslash \{r_1, \cdots, r_M\}$ together with all its connecting edges into one of its neighbors. The resulting graph will be a junction forest on $\{r_1, \cdots, r_M\}$, the maximal elements of R. □

Interestingly, the converse to this is also true:

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

Proposition 9. *If the maximal elements* $\{r_1, \cdots, r_M\}$ *can be put on a junction tree then* S_R *has no loops.*

Proof. First note that the existence of a junction tree on the maximal elements of R is equivalent to the existence of a junction tree on R. To prove the proposition we will use some results from Section 4 of (Aji and McEliece, 2000). We define a *local domain graph* G_{LD} as a weighted complete graph with vertices corresponding to the regions $r \in R$, with the weight of the edge (r, s) defined as

$$w_{(r,s)} = |r \cap s|$$

Then from Theorem 4.1 of (Aji and McEliece, 2000), any junction tree on R must be a maximal-weight spanning tree of G_{LD}. A maximal-weight spanning tree can be obtained using a greedy algorithm such as Kruskal's algorithm, see e.g. (Cormen et al., 2001): Start with an empty graph, H. Identify an edge of G_{LD} with the largest weight whose addition does not create a cycle in graph H, and add that edge to H. Repeat the preceding step until no more edges can be added.

Let G be the undirected version of the Hasse graph G_R, where each directed edge $(r \to s)$ is replaced by an undirected edge (r, s). Notice that at each stage of the above algorithm, one can choose an edge from the edge-set of G. To see this, suppose (t, r) is an edge in G_{LD} with maximal weight whose addition does not create a cycle. Then, since G is totally connected, there is a path in G where each vertex on the path contains $t \cap r$; every edge in this path then has weight at least $|t \cap r| \geq w_{t,r}$. Now note that there must be an edge (t', r') in this path (in G), such that t' and r' are not connected in H (otherwise t and r would already be connected in H and so the addition of the edge (t, r) would create a cycle.) Therefore at this stage of the algorithm we can choose the edge (t', r') instead of (t, r). This shows that there is a junction tree on R whose edge-set is a subset of the edge-set of G.

Next note that the junction tree on R constructed in the preceding paragraph, under the hypothesis that a junction tree exists on R, must include at least one edge from each EER class of G. Specifically let u_0 be a parent of r in R, and let $(u_0, u_1, u_2, \cdots, u_m = r)$ be any path between u_0 and r in G such that each u_i includes r. Then it is easy to see that $(u_0 \to r)$ and $(u_{m-1} \to r)$ must be in the same EER class. This proves that any path between u_0 and r in G includes one of the edges in the same EER class with $(u_0 \to r)$, and therefore any junction subtree of G must have at least one edge from each EER class. Now remember that S_R was defined to be a graph with precisely one edge remaining from each EER class, so we could have chosen S_R as a subgraph of the junction tree we constructed. However, if S_R has a loop, it cannot be a subgraph of a tree. □

Recall that in Section 3.2 we originally defined the Kikuchi problem in terms of a collection R_0 of regions.

Definition. 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 Propositions 8 and 9 and the definition above, 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 4(a), but can be easily handled as a junction tree 4(b).

Also, in the example of Figure 3, even though 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,

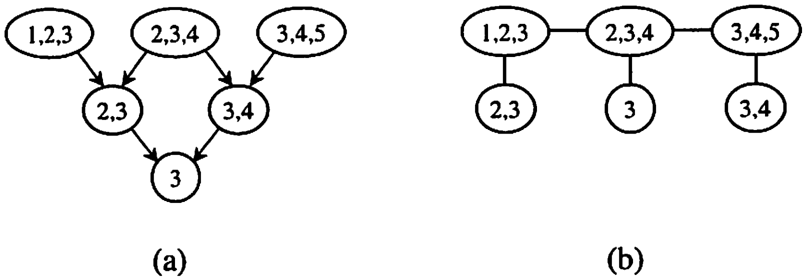


Figure 4: Hasse Diagram vs. Junction Tree

which converges to yield the solution for the Kikuchi approximation (12) which is identical to the exact marginals. In fact in these examples, the Kikuchi free energy functionalequals the variational free energy. The results in this subsection are aimed at making these observations precise.

Let R be an arbitrary collection of regions. Note that the following lemma does *not* assume that the poset R is totally connected.

Lemma 10. *If S_R is a tree, then the overcounting factor for each region $r \in R$ satisfies*

$$c_r = 1 - |\mathcal{P}_{S_R}(r)|,$$

where $|\mathcal{P}_{S_R}(r)|$ is the number of parents of r in S_R, as defined earlier in Section 4. Furthermore, the sum of the overcounting factors of all regions equals 1.

Proof. We will show this by induction on the maximum depth of regions of R.

If R has maximum depth of 0, and given that S_R is a tree, R must necessarily consist only of one region. Then the claim then holds immediately for the single region of R.

Now suppose the lemma holds for all posets R' with maximum depth l. Suppose a poset R has maximum depth $l+1$ and S_R is a tree. Let $\{(s_1^1 \rightarrow r), \cdots, (s_{k_1}^1 \rightarrow r)\}, \cdots, \{(s_1^m \rightarrow r), \cdots, (s_{k_m}^m \rightarrow r)\}$ be all the EER classes of a given region $r \in R$. Let $T_1 := \cup_{i=1}^{k_1} s_i^1, \cdots, T_m := \cup_{i=1}^{k_m} s_i^m$ be the parents of r corresponding to each EER class. Then $\mathcal{F}(T_1), \cdots, \mathcal{F}(T_m)$ must be disjoint or else two of the EER classes could be merged into a bigger class. Then by induction hypothesis, the sum of overcounting factors for each sub-poset $\mathcal{F}(T_1), \cdots, \mathcal{F}(T_m)$ is 1, and by

Lemma 6 these are the same as the overcounting factors w.r.t. R. Then

$$c_r = 1 - \sum_{s \in \mathcal{A}(r)} c_s = 1 - \sum_{i=1}^{m} \sum_{u \in T_i} c_u = 1 - \sum_{i=1}^{m} 1 = 1 - |\mathcal{P}_{S_R}(r)|$$

since the number of parents of r in S_R is precisely the number of EER classes of r.

Now consider the sum of the overcounting factors of all nodes:

$$\begin{aligned}\sum_{r \in R} c_r &= \sum_{r \in R} (1 - |\mathcal{P}_{S_R}(r)|) \\ &= \sum_{r \in R} 1 - \sum_{(s \to r) \in \mathcal{E}(S_R)} 1 \\ &= |V(S_R)| - |\mathcal{E}(S_R)| \\ &= 1\end{aligned}$$

where the last equality follows from the fact that S_R is assumed to be a tree, so the number of its vertices is one more than the number of its edges. This completes the inductive step of the proof. □

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

Theorem 11. *(Exactness of Kikuchi approximates, 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 $b(\mathbf{x}) = \prod_{r \in R} b_r(\mathbf{x}_r)^{c_r} \ \forall \mathbf{x}$

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) towards 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 (6) is exact. Now

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

□

Corollary 12. *If R_0 is loop-free, then the constrained minimization problem* (12) *has a unique solution. Further, the solutions $\{b_r^*,\, r \in R\}$ is the exact marginals of the product function, and the minimum free energy equals the log-partition function, i.e. $b_r^*(\mathbf{x}_r) = B_r(\mathbf{x}_r)$ and $F^* = -\log(Z)$.* [7]

Proof. From Theorem 11, if S_R has no loops then $\Delta_R^K = \Delta_R$, and further for each $\{b_r\} \in \Delta_R^K$, the function $\prod_{r\in R} b_r^{c_r}(\mathbf{x}_r)$ is a valid distribution on $\mathbf{x}$ which marginalizes to the b_r's, and therefore $F_R^K(\{b_r\}) = F(\prod_{r\in R} b_r^{c_r})$. On the other hand, as stated in Section 3.1, the minimum of $F(b)$ is achieved uniquely with Boltzmann distribution, $B(\mathbf{x}) := \frac{1}{Z}\prod_{r\in R} \alpha_r(\mathbf{x}_r)$. Now clearly $\{B_r,\ r \in R\} \in \Delta_R = \Delta_R^K$, since $\{B_r,\, r \in R\}$ is the set of R-marginals of a valid distribution $B(\mathbf{x})$. Therefore $\{B_r,\ r \in R\}$ is a minimizer of (12), and the corresponding minimum F^* indeed equals the minimum F_0 of $F(b)$, which in turn is equal to $-\log(Z)$.

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

To see the uniqueness, note that if $\{b_r^*,\ r \in R\} \in \Delta_R^K$ is any minimizer of (12), then the corresponding distribution $b^*(\mathbf{x}) := \prod_{r\in R} b_r^{*c_r}(\mathbf{x}_r)$ is a distribution, with marginals $\{b_r^*\}$, which minimizes the variational free energy $F(b)$; but $B(\mathbf{x})$ is the unique minimizer of $F(b)$. Therefore $b^*(\mathbf{x})$ must be equal to $B(\mathbf{x})$, and accordingly $\{b_r^*\} = \{B_r\}$. □

The above results show that a sufficient condition for the exactness of the solutions of Kikuchi approximation method of (12) 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 (6) exact for all R-decomposable distributions?* The following theorem, answers this question.

Theorem 13. *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 collection $\{k_r\,,\ r \in R\}$ exists, then it is unique and equals the (Möbius) overcounting factors $\{c_r\,,\ r \in R\}$.

Proof. Suppose (2) does not hold, so for some $s \subseteq [N]$ such that $\cup_{i\in s}\mathcal{F}(i)$ is connected, $\sum_{r\in\cup_{i\in s}\mathcal{F}(i)} k_r \neq 1$. We will choose kernels $\{\alpha_r,\ r \in R\}$ so that (1) will be violated.
Specifically, for each r we choose $\alpha_r(\mathbf{x}_r) = \prod_{j\in r\setminus s} 1(x_j = 0)\big(\prod_{i\in r\cap s} 1(x_i = 0) + \prod_{i\in r\cap s} 1(x_i = 1)\big)$. Under the product distribution $B(\mathbf{x}) = \frac{1}{Z}\prod_{r\in R}\alpha_r(\mathbf{x}_r)$, for each $j \in [N]\setminus s$, x_j will have zero probability of taking a value other than 0, i.e. random variable x_j is deterministic and will have zero entropy. On the other hand, since $\cup_{i\in s}\mathcal{F}(i)$ is connected, for each pair $i, j \in s$ the probability that $x_i \neq x_j$ is

zero; in fact there is exactly a probability of 0.5 that $\mathbf{x}_s = (0, 0, \cdots, 0)$ and a probability of 0.5 that $\mathbf{x}_s = (1, 1, \cdots, 1)$. Therefore random variables x_i for $i \in s$ are redundant, and for all $t \subseteq [N]$ s.t. $s \cap t \neq \emptyset$, $H_t(B_t) = 1$(bit). Now by independence of $\mathbf{x}_s$ and $\mathbf{x}_{[N]\backslash s}$ we have

$$H(B) = H_s(B_s) + H_{[N]\backslash s}(B_{[N]\backslash s}) = H_s(B_s) = 1$$

On the other hand,

$$\sum_{r \in R} k_r\, H_r(B_r) = \sum_{r \in \cup_{i \in s} \mathcal{F}(i)} k_r\, H_r(B_r) + \sum_{r \in R \backslash \cup_{i \in s} \mathcal{F}(i)} k_r\, H_r(B_r) = \sum_{r \in \cup_{i \in s} \mathcal{F}(i)} k_r \cdot 1 + 0 \neq 1$$

so that $H(B) \neq \sum_{r \in R} k_r\, H_r(B_r)$. Therefore we have shown that (1) implies (2).

Now suppose that (2) holds. We will show that (3) must hold, using induction on the *lexicographical order on the strings of the decreasingly-sorted cardinalities of elements of* S defined on all $S \subseteq 2^{[N]}$; we clarify this ordering using an example:

Suppose $N = 12$, and $S_1 = \{\{10, 11, 0\}, \{1, \cdots, 10\}\}$, $S_2 = \{\{1, 2, 3, 4, 5\}, \{3, 4, 5, 6\}, \{6, 7, 8, 9, 10\}\}$, $S_3 = \{\{1\}, \cdots, \{7\}\}$ and $S_4 = \{\{6\}, \{7\}, \{8\}\}$. Then the 'sorted strings of the cardinalities' are $str(S_1) = [10.3]$, $str(S_2) = [5.5.4]$, $str(S_3) = [1.1.1.1.1.1.1]$ and $str(S_4) = [1.1.1]$, so that $str(S_1) >_{lex} str(S_2) >_{lex} str(S_3) >_{lex} str(S_4)$.

It is clear that if all $s \in S$ were singletons, so that $str(S) = [1. \cdots .1]$, then (3) is equivalent to (2). Now suppose $S = \{s_1, \cdots, s_n\}$, and $|s_n| \geq 2$. We split s_n as the disjoint union of t_1 and t_2, i.e. $s_n = t_1 \cup t_2$ and $t_1 \cap t_2 = \emptyset$, so that $0 < |t_1|, |t_2| < |s_n|$. Define $T_1 := \{s_1, \cdots, s_{n-1}, t_1\}, T_2 := \{s_1, \cdots, s_{n-1}, t_2\}$ and $T_{12} := \{s_1, \cdots, s_{n-1}, t_1, t_2\}$. Clearly now, with the above lexicographical order, $str(T_1), str(T_2)$ and $str(T_{12})$ each are smaller than $str(S)$. Furthermore, $\cup_{s \in T_1} \mathcal{F}(s), \cup_{s \in T_2} \mathcal{F}(s)$ and $\cup_{s \in T_{12}} \mathcal{F}(s)$ are each connected, since they all contain $\cup_{s \in S} \mathcal{F}(s)$ as an up-set. But

$$\begin{aligned}
\sum_{r \in \cup_{s \in T_{12}} \mathcal{F}(s)} k_r &= \sum_{r \in \cup_{s \in S} \mathcal{F}(s)} k_r + \sum_{r \in \mathcal{F}(t_1) \backslash \cup_{s \in S} \mathcal{F}(s)} k_r + \sum_{r \in \mathcal{F}(t_2) \backslash \cup_{s \in S} \mathcal{F}(s)} k_r &= 1 \\
\sum_{r \in \cup_{s \in T_1} \mathcal{F}(s)} k_r &= \sum_{r \in \cup_{s \in S} \mathcal{F}(s)} k_r + \sum_{r \in \mathcal{F}(t_1) \backslash \cup_{s \in S} \mathcal{F}(s)} k_r &= 1 \\
\sum_{r \in \cup_{s \in T_2} \mathcal{F}(s)} k_r &= \sum_{r \in \cup_{s \in S} \mathcal{F}(s)} k_r + \sum_{r \in \mathcal{F}(t_2) \backslash \cup_{s \in S} \mathcal{F}(s)} k_r &= 1
\end{aligned}$$

where we have used induction hypothesis to conclude that each sum must be equal to 1. Using the above three equations we get $\sum_{r\in\cup_{s\in S}\mathcal{F}(s)} k_r = 1$. This completes the inductive proof.

Next suppose that (3) holds for a choice of factors $\{k_r, r \in R\}$. First note that choosing $S = \{r\}$ for each $r \in R$ we get equations (7), implying that $\{k_r, r \in R\}$ must in fact be the same as the (Möbius) overcounting factors, $\{c_r, r \in R\}$. Suppose now that S_R, the minimal graph of R has a loop. Let $L \subseteq R$ be a loop of S_R, and let $L_0 \subset R$ be the collection of minimal regions of L, i.e. every $r \in L$ contains some $r_0 \in L_0$, and that no region in L_0 properly contains another region in L_0. Therefore $\mathcal{F}(L_0)$ contains loop L of S_R. We now claim that one can find a loop L with minimal regions L_0 such that for any proper subset $L_0' \subset L_0$, $\mathcal{F}(L_0')$ is loop-free. This is because if $\mathcal{F}(L_0')$ contains a loop L' for a proper subset L_0' of L_0, then we can choose L_0' in place of L_0, and $\mathcal{F}(L_0')$ still has a loop L'. But $|L_0|$ is finite and $|L_0'| < |L_0|$, so this process must end, yielding a loop L with collection L_0 of minimal regions, with the desired property. Further note that L_0 cannot have cardinality 1, since if $L_0 = \{r_0\}$ for some $r_0 \in R$, then all the edges of the Hasse diagram that terminate in r_0 and participate in the loop L would be EER; all but one of these edges would be removed in S_R, therefore r_0 cannot be part of a loop.

Therefore for each region $r \in \mathcal{F}(L_0)$, $S_{\mathcal{F}(r)}$ is loop-free. Noting that the overcounting factor c_r only depends on $\mathcal{F}(r)$, and using Lemma 10, $c_r = 1 - |\mathcal{P}_{S_R}(r)|$. Then, as before, the sum $\sum_{r\in\mathcal{F}(L_0)} c_r$ can be rewritten as the difference between the number of vertices and the number of edges of $S_{\mathcal{F}(L_0)}$. But $S_{\mathcal{F}(L_0)}$ has at least one loop, therefore it has at least as many edges as vertices. Therefore $\sum_{r\in\mathcal{F}(L_0)} c_r \leq 0$ and cannot be equal to 1. This would contradict (3), and hence S_R must be loop-free.

Now suppose that S_R is loop-free. Then by Proposition 8, S_R is a junction tree. Choose then $\{k_r\}$ to be equal to the overcounting factors $\{c_r\}$, so that $k_r = 1 - |\mathcal{P}_{S_R}(r)|$. Then by standard results on the junction trees, any distribution $B(\mathrm{x})$ that decomposes on the junction tree S_R, factors as $\prod_{r\in R} B_r(\mathrm{x}_r)^{k_r}$ (see (Cowell et al., 1999)). From this (1) follows immediately. This completes the proof of the theorem. □

From this theorem, proof of Proposition 1 of Section 3.1 follows:

Proof of Proposition 1: Note that although the statement of Theorem 13 assumes that R is totally connected, that assumption is only needed to show that (3) implies (1). From the proof of Theorem 13, for an arbitrary collection of regions R, any collection of factors $\{k_r, r \in R\}$ satisfying condition (1) of Theorem 13 must be the Möbius overcounting factors. □

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.

Theorem 14. *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* (12) *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 (6) is exact for the Boltzmann distribution. Specifically, define

$$T := \{\ \{\alpha_r^0(\mathbf{x}_r),\ r \in R_0\} : \sum_{\mathbf{x}} B(\mathbf{x}) \log(B(\mathbf{x})) = \sum_{r \in R} c_r \sum_{\mathbf{x}_r} B_r(\mathbf{x}_r) \log(B_r(\mathbf{x}_r))\ \} \tag{19}$$

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 (6), i.e.

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

Then $T = f^{-1}(0)$. Function $f(\cdot)$ above is clearly analytic on its domain, $\mathbb{R}^l_+$. Then, as demonstrated in (Federer, 1969) §3.1.24, 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 13 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 be derived. Although we have so far focused on positive distributions $B(\mathbf{x}) > 0$, in many applications one is interested in distributions that can be zero at certain points of the state space. As mentioned above, using its continuity, the function $x\log(x)$ can be extended conveniently at the point $x = 0$. This means that we can handle arbitrary (not necessarily positive) distributions in the above framework.

Suppose then we condition on the collections of kernels $\{\alpha^0_r(\mathbf{x}_r), r \in R_0\}$ which are zero at predetermined values. Specifically, for each $r \in R_0$, let $Z_r \subset \prod_{i\in r}[q_i]$ be a subset of the range of values of $\mathbf{x}_r$, on which α^0_r is zero. We say a collection of kernels $\{\alpha^0_r(\mathbf{x}_r), r \in R_0\}$ is *consistent* with $\{Z_r, r \in R_0\}$ if for each $r \in R_0$, $\alpha^0_r(\mathbf{x}_r) = 0$ for all $\mathbf{x}_r \in Z_r$.

Theorem 15. *Suppose R_0 is loopy, and let $\{Z_r, r \in R\}$ be a collection of zeros of the kernels as defined above, chosen such that the following holds:*

$$\forall\, s \subseteq [N],\ \exists\, \tilde{\mathbf{x}}^1_s, \tilde{\mathbf{x}}^2_s, \tilde{\mathbf{x}}_{[N]\setminus s} \text{ such that } \tilde{x}^1_i \neq \tilde{x}^2_i\ \forall i \in s, \text{ and that } B(\tilde{\mathbf{x}}^1_s, \tilde{\mathbf{x}}_{[N]\setminus s}) \text{ and } B(\tilde{\mathbf{x}}^2_s, \tilde{\mathbf{x}}_{[N]\setminus s}) \text{ can both be nonzero under the restrictions imposed by } \{Z_r\}. \tag{20}$$

Then the conditional measure of the set of kernels $\{\alpha^0_r(\mathbf{x}_r), r \in R_0\}$ conditioned to be consistent with $\{Z_r, r \in R_0\}$, for which Kikuchi approximation method of (12) *is exact is zero.*

Proof. We define the set T and error function $f(\cdot)$ similar to the proof of Theorem 14, with the convention that $0\log(0) = 0$. Here, however, f is a function on $\mathbb{R}^l_+$ with $l = \sum_{r\in R_0}(q_r - |Z_r|)$. Once again, $f(\cdot)$ is seen to be analytic on $\mathbb{R}^l_+$.

The argument to show that f is not identically zero proceeds exactly as before. It only remains to show that Theorem 13 still holds for the restricted case imposed by $\{Z_r\}$. The proof of Theorem 13 remains unchanged with the following exception: in the first part, to prove that "not (2) implies not (1)", for each r we choose $\alpha_r(\mathbf{x}_r) = \prod_{j \in [N]\setminus(r\cap s)} 1(x_j = \tilde{x}_j)\big(\prod_{i \in r\cap s} 1(x_i = \tilde{x}_i^1) + \prod_{i\in r\cap s} 1(x_i = \tilde{x}_i^2)\big)$, where $\tilde{\mathbf{x}}_s^1, \tilde{\mathbf{x}}_s^2$ and $\tilde{\mathbf{x}}_{[N]\setminus s}$ are chosen according to (20). The fact that these α's are consistent with $\{Z_r\}$ is then guaranteed by (20).

Therefore f is not identically zero on its domain $\mathbb{R}_+^l$ of kernels consistent with $\{Z_r\}$, and hence the Lebesgue measure of the set T is zero. □

We conclude this section with a generalization of Corollary 4 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 17 for convexity of Kikuchi free energy.

Definition. We call a collection R of Kikuchi regions *normal,* if for all $S \subset R$, there exist a largest region (w.r.t. set-inclusion) $m_S \in \bigcap_S \mathcal{D}(s)$, possibly the empty set, that contains any other region $u \in \bigcap_S \mathcal{D}(s)$. □

Lemma 16. *If R is normal and S_R is connected and has exactly one loop, then* $\sum_{r\in R} c_r \in \{0, 1\}$.

Proof. Let $L \subseteq R$ be the set of nodes in the single loop of S_R, and let $\widetilde{L} := \mathcal{F}(L)$. Note that L cannot have a single minimum region, i.e. a region that is contained in all other regions of L; to see this, suppose to the contrary that r_0 is the minimum region of L. Then all the edges of the Hasse diagram that terminate in r_0 and participate in the loop L would be EER; all but one of these edges would be removed in S_R, therefore r_0 could not be part of the loop L, which is a contradiction.

It follows from this that for each $r \in \widetilde{L}$, the sub-poset $\mathcal{F}(r)$ does not contain the loop L, and is hence loop-free. Then by Lemmas 10 and 6, $c_r = 1 - |\mathcal{P}_{S_{\mathcal{F}(r)}}(r)| = 1 - |\mathcal{P}_{S_{\widetilde{L}}}(r)|$. Then, as argued in the proof of Lemma 10, there is a contribution of $+1$ for each vertex and -1 for each edge of $S_{\widetilde{L}}$ for the sum $\sum_{s\in\widetilde{L}} c_s$. Now $S_{\widetilde{L}}$ is connected and has one loop, so the number of its vertices equal the number of its edges and so $\sum_{s\in\widetilde{L}} c_s = 0$.

Now for each $r \in R\setminus\widetilde{L}$ such that $S_{\widetilde{L}\cup\mathcal{F}(r)}$ is connected, we calculate the contribution of the overcounting factors of regions in $\mathcal{F}(r)\setminus\widetilde{L}$ to the overall sum. After each stage, we will inductively append $\mathcal{F}(r)$ to $\widetilde{L}$.

First suppose that $\mathcal{F}(r)$ does not contain the loop L of R; then

$$\sum_{s\in\widetilde{L}\cup\mathcal{F}(r)} c_s = \sum_{s\in\widetilde{L}} c_s + \sum_{s\in\mathcal{F}(r)\setminus\widetilde{L}} c_s \tag{21}$$

We will argue that the second term must equal 0. Remember that from the definition of the overcounting factors,

$$1 = \sum_{s \in \mathcal{F}(r)} c_s = \sum_{s \in \mathcal{F}(r) \cap \tilde{L}} c_s + \sum_{s \in \mathcal{F}(r) \setminus \tilde{L}} c_s \tag{22}$$

But $S_{\mathcal{F}(r) \cap \tilde{L}}$ must be a tree, since firstly it does not contain the loop L, and secondly r can have only one parent in $S_{\mathcal{F}(r) \cap \tilde{L}}$ or else S_R would have a loop containing r. Therefore by Lemma 10, $\sum_{s \in \mathcal{F}(r) \cap \tilde{L}} c_s = 1$. Then from (22), $\sum_{s \in \mathcal{F}(r) \setminus \tilde{L}} c_s = 0$ and hence by (21), $\sum_{s \in \tilde{L} \cup \mathcal{F}(r)} c_s = \sum_{s \in \tilde{L}} c_s$, i.e. the sum is preserved after appending $\mathcal{F}(r)$.

Next suppose that $\mathcal{F}(r)$ contains the loop L of R. Then since R is normal, there is a largest $m \in R$ such that $\mathcal{F}(m)$ contains the loop. Again we break up the new sum of overcounting factors as in equation (21). If r is equal to m, $\sum_{s \in \mathcal{F}(r) \cap \tilde{L}} c_s$ in (22) equals 0 since $S_{\mathcal{F}(r) \cap \tilde{L}}$ has one loop and has no region r' such that $\mathcal{F}(r')$ contains the loop and hence by what we have shown so far, $\sum_{s \in S_{\mathcal{F}(r) \cap \tilde{L}}} c_s = 0$. Therefore by (22), $\sum_{s \in \mathcal{F}(r) \setminus \tilde{L}} c_s = 1$, and hence from (21), $\sum_{s \in \tilde{L} \cup \mathcal{F}(m)} c_s = \sum_{s \in \tilde{L}} c_s + 1 = 1$.

On the other hand, whenever $\mathcal{F}(r)$ contains the loop but $r \neq m$, then $m \subset R$ and hence $\mathcal{F}(m) \subset \tilde{L} \cap \mathcal{F}(r)$. Then by what has been shown so far, $\sum_{s \in \tilde{L} \cap \mathcal{F}(r)} c_s = 1$. Then the additional term $\sum_{s \in \mathcal{F}(r) \setminus \tilde{L}} c_s$ in (21) is 0 and hence $\sum_{s \in \tilde{L} \cup \mathcal{F}(r)} c_s = \sum_{s \in \tilde{L}} c_s = 1$.

Therefore we have shown that, depending on whether $\bigcap_{s \in L} \mathcal{D}(s)$ is empty or not, the sum $\sum_{r \in R} c_r$ is 0 or 1. □

Theorem 17. *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 Variational Method of (Yedidia et al., 2001) is strictly convex if S_R has zero or one loop.*

Proof. For each $S \subset R$, $\mathcal{F}(S)$ contains zero or one loop. Then by Lemmas 10 and 16, $\sum_{s \in \mathcal{F}(S)} c_s \geq 0$ and hence, from Theorem 3 the Kikuchi functional is strictly convex. □

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 (12). Previously described algorithms such as the generalized belief propagation (GBP) algorithms of (Yedidia et al., 2001) and (Yedidia et al., 2002), and Poset-BP algorithm of (McEliece and Yildrim, 2003) are special cases of the class of algorithms we describe. The algorithms proposed earlier work on the full Hasse diagram. The results derived in the earlier sections of this paper on the minimal graphs allow us to propose algorithms for solving (12) which 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 on 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 (12).

Now let G be any graphical representation of Δ_R^K as defined in Section 4. Then the Lagrangian of equation (13) 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 edge-constraints defined in Section 4. Motivated by an observation made by Yedidia, Freeman and Weiss in (Yedidia et al., 2001; Yedidia et al., 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 (12).

Definition. The *YFW edge-constraint*[8] for an edge $(s \to t)$ of G is defined as the

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

following functional of the pseudo-marginals $\{b_r\,,\, r \in R'\}$:

$$\mathrm{EC}'_{(s\to t)}(\{b_r\,,\, r \in R'\}) := \sum_{u\in\mathcal{F}(t)\setminus\mathcal{F}(s)} c_u \sum_{\mathbf{x}_{u\setminus t}} b_u(\mathbf{x}_u) \tag{23}$$

where $R' := \{r \in R,\ c_r \neq 0\}$ is the collection of regions with non-zero 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 (23). When the arguments are clear from the context, we abbreviate these edge-constraints as $\mathrm{EC}'_{(s\to t)}$. □

Proposition 18. *The collection of pseudo-marginals represented by the YFW edge-constraints is equal to the restriction of* Δ_R^K *to* R'. *Namely, if we define*

$$\Delta'_R := \big\{\, \{b_r(\mathbf{x}_r),\ r \in R'\}\ :\ \forall (s \to t) \in \mathcal{E}(G),\ \mathrm{EC}'_{(s\to t)}(\{b_r, r \in R'\}) = 0 \\ \text{and } \forall r \in R',\ \sum_{\mathbf{x}_r} b_r(\mathbf{x}_r) = 1 \big\}, \tag{24}$$

then $\Delta'_R = \Delta_R^K\big|_{R'}$, *where*

$$\Delta_R^K\big|_{R'} := \big\{\, \{b_r(\mathbf{x}_r), r \in R'\} : \{b_r(\mathbf{x}_r), r \in R'\} \text{ has an extension } \{b_r(\mathbf{x}_r), r \in R\} \in \Delta_R^K \,\big\}$$

is the restriction of Δ_R^K *to* R'.

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

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

$$\text{Therefore} \sum_{u\in\mathcal{F}(t)\setminus\mathcal{F}(s)} c_u = 0 \tag{25}$$

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 $\Delta^K_R\big|_{R'} \subseteq \Delta'_R$.

Now conversely suppose that $\{b_r,\, r \in R'\} \in \Delta'_R$. We will show by induction on depth function $d(t)$ of region $t \in R$ (w.r.t. 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 vacuously for the maximal regions, since these regions cannot have parents. Now let t be a region with depth $d(t) = l > 0$ and let $\mathcal{P}_G(t) = \{s_1, \cdots, s_m\}$. 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:

  $$\sum_{u \in \mathcal{F}(s_i)} c_u \sum_{\mathbf{x}_{u\setminus t}} b_u(\mathbf{x}_u) = \sum_{u \in \mathcal{F}(s_j)} c_u \sum_{\mathbf{x}_{u\setminus t}} b_u(\mathbf{x}_u) \qquad (26)$$

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

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

$$\begin{aligned} & \sum_{u \in \mathcal{F}(t) \setminus \mathcal{F}(s_i)} c_u \sum_{\mathbf{x}_{u \setminus t}} b_u(\mathbf{x}_u) = 0 \\ \Longrightarrow \;& c_t\, b_t(\mathbf{x}_t) + \sum_{u \in \mathcal{A}(t) \setminus \mathcal{F}(s_i)} c_u\, b'_t(\mathbf{x}_t) = 0 \\ \Longrightarrow \;& b_t(\mathbf{x}_t) = b'_t(\mathbf{x}_t) \end{aligned}$$

since by (25), $c_t + \sum_{u \in \mathcal{A}(t) \setminus \mathcal{F}(s_i)} c_u = 0$, and $c_t \neq 0$.

So 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 7 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, i.e. $\sum_{\mathbf{x}_{u \setminus t}} b_u(\mathbf{x}_u) = b_t(\mathbf{x}_t)$.
Therefore we have found the desired extension $\{b_r,\, r \in R\} \in \Delta_R$, and so $\Delta'_R \subseteq \Delta_R^K|_{R'}$. This proves that $\Delta'_R = \Delta_R^K|_{R'}$ as claimed. □

Remark. Note from (9) and (11) that Kikuchi free energy can be rewritten as follows:

$$F_R^K(\{b_r(\mathbf{x}_r)\}) = \sum_{r \in R} \sum_{\mathbf{x}_r} \big(- c_r\, b_r(\mathbf{x}_r) \log(\beta_r(\mathbf{x}_r)) + c_r\, b_r(\mathbf{x}_r) \log(b_r(\mathbf{x}_r)) \big) \qquad (27)$$

From (27) it is apparent that $F_R^K(\{b_r, r \in R\})$ only depends on $\{b_r, r \in R'\}$, since the terms involving the pseudo-marginals corresponding to the regions with zero overcounting factors are multiplied by zero. Therefore (12) can be rewritten as follows

$$\begin{aligned} \min_{\{b_r, r\in R\}\in\Delta_R^K} F_R^K(\{b_r, r\in R\}) &= \min_{\{b_r, r\in R'\}\in\Delta'_R} F_R^K(\{b_r, r\in R'\}) \\ \text{and} \quad \Big(\arg \min_{\{b_r, r\in R\}\in\Delta_R^K} F_R^K(\{b_r, r\in R\})\Big)\Big|_{R'} &= \arg \min_{\{b_r, r\in R'\}\in\Delta'_R} F_R^K(\{b_r, r\in R'\}) \end{aligned} \tag{28}$$

In other words, the central constrained minimization problem (12) is reduced to the following: $\min_{\{b_r, r\in R'\}\in\Delta'_R} F_R^K(\{b_r, r\in R'\})$. □

We will now write the Lagrangian for (28) using the YFW edge-constraints:

$$\begin{aligned} \mathcal{L} := &\sum_{r\in R} c_r\, b_r(\mathbf{x}_r) \log\big(\frac{b_r(\mathbf{x}_r)}{\beta_r(\mathbf{x}_r)}\big) \\ &+ \sum_{(r\to t)\in\mathcal{E}(G)} \sum_{\mathbf{x}_t} \lambda_{rt}(\mathbf{x}_t) \sum_{u\in\mathcal{F}(t)\setminus\mathcal{F}(r)} c_u \sum_{\mathbf{x}_{u\setminus t}} b_u(\mathbf{x}_u) + \sum_{r\in R} \kappa_r \big(\sum_{\mathbf{x}_r} b_r(\mathbf{x}_r) - 1\big) \end{aligned} \tag{29}$$

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:

$$b_r(\mathbf{x}_r) = k\,\beta_r(\mathbf{x}_r)\Big(\prod_{p\in\mathcal{P}_G(r)} m_{pr}(\mathbf{x}_r)\Big)\Big(\prod_{d\in\mathcal{D}(r)} \prod_{p'\in\mathcal{P}_G(d)\setminus(\{r\}\cup\mathcal{D}(r))} m_{p'd}(\mathbf{x}_d)\Big) \tag{30}$$

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 $\sum_{\mathbf{x}_{p\setminus r}} b_p(\mathbf{x}_p) - b_r(\mathbf{x}_r) = 0$:

$$m_{pr}(\mathbf{x}_r) = k'\frac{\sum_{\mathbf{x}_{p\setminus r}} \beta_p(\mathbf{x}_p)\Big(\prod_{s\in\mathcal{P}_G(p)} m_{sp}(\mathbf{x}_p)\Big)\Big(\prod_{d\in\mathcal{D}(p)} \prod_{s'\in\mathcal{P}_G(d)\setminus(\{p\}\cup\mathcal{D}(p))} m_{s'd}(\mathbf{x}_d)\Big)}{\beta_r(\mathbf{x}_r)\Big(\prod_{s\in\mathcal{P}_G(r)\setminus\{p\}} m_{sr}(\mathbf{x}_r)\Big)\Big(\prod_{d\in\mathcal{D}(r)} \prod_{p'\in\mathcal{P}_G(d)\setminus(\{r\}\cup\mathcal{D}(r))} m_{p'd}(\mathbf{x}_d)\Big)} \tag{31}$$

where k' is any convenient constant. Note that the common terms from the numerator and denominator of (31) can be cancelled, but to avoid even longer formulas we will not write the explicit form here.

The fixed points of equations (30) and (31) 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 (30) and (31) is defined on any graphical representation of Δ_R^K, and has as many messages as the edges of the underlying graph. From results of Section 4 then, 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. A version of this algorithm was originally labeled GBP in (Yedidia et al., 2001). In (McEliece and Yildrim, 2003) also the authors described an algorithm called 'Poset-BP' which is equivalent to the restriction of our results when G is the Hasse diagram. Our result shows that in general there are algorithms with strictly fewer messages, 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. Not only the messages corresponding to the removed edges need not be updated at each iteration of the algorithm, the update rules for the remaining messages are also less complex, since they depend on fewer edges.

It is also noteworthy that the proofs 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 pseudo-marginals at other regions (see e.g. the poset in Figure 5). We have avoided this restriction, by proving the results for a general poset. □

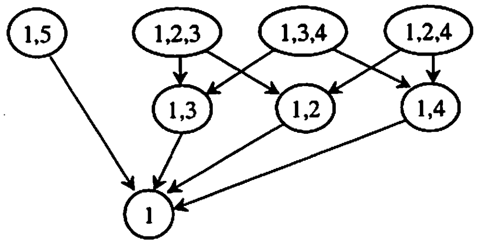


Figure 5: Region '1' has zero overcounting, but cannot be removed.

Consider now the restriction of the above algorithm in the Bethe case, i.e.

each region in R is either maximal or minimal w.r.t. 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 addition to the messages $m_{pr}(\mathbf{x}_r)$ for $r \subset p$, we also define messages from a child to parent as follows:

$$n_{rp}(\mathbf{x}_r) := \beta_r(\mathbf{x}_r) \prod_{s \in \mathcal{P}(r) \setminus \{p\}} m_{sr}(\mathbf{x}_r) \tag{32}$$

Then, by equation (30), for a maximal region $p \in R$,

$$b_p(\mathbf{x}_p) = k\, \beta_p(\mathbf{x}_p) \prod_{d \in \mathcal{D}(p)} n_{pd}(\mathbf{x}_d) \tag{33}$$

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

$$b_r(\mathbf{x}_r) = k\, \beta_r(\mathbf{x}_r) \prod_{d \in \mathcal{P}(r)} m_{dr}(\mathbf{x}_r) \tag{34}$$

The update equation (31) for messages m_{pr} can then be rewritten as

$$\begin{aligned} m_{pr}(\mathbf{x}_r) &= k' \frac{\sum_{\mathbf{x}_{p \setminus r}} \beta_p(\mathbf{x}_p) \prod_{d \in \mathcal{D}(p)} n_{dp}(\mathbf{x}_d)}{\beta_r(\mathbf{x}_r)\, n_{rp}(\mathbf{x}_r)} \\ &= k' \sum_{\mathbf{x}_{p \setminus r}} \frac{\beta_p(\mathbf{x}_p)}{\beta_r(\mathbf{x}_r)} \prod_{d \in \mathcal{D}(p) \setminus \{r\}} n_{dp}(\mathbf{x}_d) \end{aligned} \tag{35}$$

It is now easy to see that equations (32)–(35) precisely define the conventional belief propagation algorithm of (Pearl, 1988) applied on G_R.

Example 4. Consider a poset $R = \{r, s, t, u, v, w\}$ with the Hasse diagram G_R given in Figure 6(a). We will write the explicit form the GBP algorithm on both G_R and S_R.
GBP on G_R:

Messages:

$$m_{ru}(\mathbf{x}_u) = \frac{\sum_{\mathbf{x}_{r \setminus u}} \beta_r(\mathbf{x}_r)}{\beta_u(\mathbf{x}_u)}, \qquad m_{tv}(\mathbf{x}_v) = \frac{\sum_{\mathbf{x}_{t \setminus v}} \beta_t(\mathbf{x}_t)}{\beta_v(\mathbf{x}_v)}$$

$$m_{su}(\mathbf{x}_u) = \frac{\sum_{\mathbf{x}_{s \setminus u}} \beta_s(\mathbf{x}_s)\, m_{tv}(\mathbf{x}_v)}{\beta_u(\mathbf{x}_u)\, m_{vw}(\mathbf{x}_w)}, \qquad m_{sv}(\mathbf{x}_v) = \frac{\sum_{\mathbf{x}_{s \setminus v}} \beta_s(\mathbf{x}_s)\, m_{ru}(\mathbf{x}_u)}{\beta_v(\mathbf{x}_v)\, m_{uw}(\mathbf{x}_w)}$$

$$m_{uw}(\mathbf{x}_w) = \frac{\sum_{\mathbf{x}_{u \setminus w}} \beta_u\, m_{ru}\, m_{su}}{\beta_w(\mathbf{x}_w)}, \qquad m_{vw}(\mathbf{x}_w) = \frac{\sum_{\mathbf{x}_{v \setminus w}} \beta_v\, m_{tv}\, m_{sv}}{\beta_w(\mathbf{x}_w)}$$

Beliefs:

$$b_r(\mathbf{x}_r) = \beta_r(\mathbf{x}_r)\, m_{su}(\mathbf{x}_u)\, m_{vw}(\mathbf{x}_w), \qquad b_t(\mathbf{x}_t) = \beta_t(\mathbf{x}_t)\, m_{sv}(\mathbf{x}_v)\, m_{uw}(\mathbf{x}_w)$$

$$b_s(\mathbf{x}_s) = \beta_s(\mathbf{x}_s)\, m_{ru}(\mathbf{x}_u)\, m_{tv}(\mathbf{x}_v), \qquad b_u(\mathbf{x}_u) = \beta_u(\mathbf{x}_u)\, m_{ru}(\mathbf{x}_u)\, m_{su}(\mathbf{x}_u)\, m_{vw}(\mathbf{x}_w)$$

$$b_w(\mathbf{x}_w) = \beta_w(\mathbf{x}_w)\, m_{uw}(\mathbf{x}_w)\, m_{vw}(\mathbf{x}_w), \qquad b_v(\mathbf{x}_v) = \beta_v(\mathbf{x}_v)\, m_{sv}(\mathbf{x}_v)\, m_{tv}(\mathbf{x}_v)\, m_{uw}(\mathbf{x}_w)$$

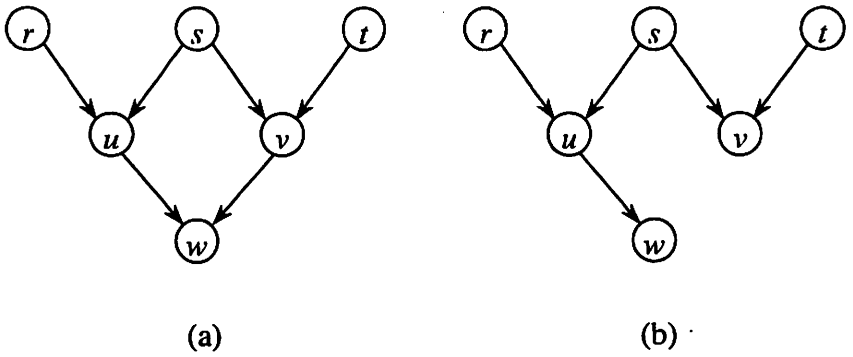


Figure 6: (a) Hasse diagram G_R, and (b) a minimal graphical representation S_R of the problem in Example 4

Note that this algorithm contains a 'loop': m_{su} depends on m_{vw}, which depends on m_{sv}, which depends on m_{uw}, which in turn depends on m_{su}. This means that the above messages will not converge in finite time, even though, as is apparent from Figure 6(b), a junction tree does exist.

Compare the above algorithm with the following:
GBP on S_R:

$$\text{Messages:}\quad m_{ru}(\mathbf{x}_u) = \frac{\sum_{\mathbf{x}_{r\setminus u}} \beta_r(\mathbf{x}_r)}{\beta_u(\mathbf{x}_u)}, \qquad m_{tv}(\mathbf{x}_v) = \frac{\sum_{\mathbf{x}_{t\setminus v}} \beta_t(\mathbf{x}_t)}{\beta_v(\mathbf{x}_v)}$$

$$m_{su}(\mathbf{x}_u) = \frac{\sum_{\mathbf{x}_{s\setminus u}} \beta_s(\mathbf{x}_s)\, m_{tv}(\mathbf{x}_v)}{\beta_u(\mathbf{x}_u)}, \qquad m_{sv}(\mathbf{x}_v) = \frac{\sum_{\mathbf{x}_{s\setminus v}} \beta_s(\mathbf{x}_s)\, m_{ru}(\mathbf{x}_u)}{\beta_v(\mathbf{x}_v)\, m_{uw}(\mathbf{x}_w)}$$

$$m_{uw}(\mathbf{x}_w) = \frac{\sum_{\mathbf{x}_{u\setminus w}} \beta_u\, m_{ru}\, m_{su}}{\beta_w(\mathbf{x}_w)}$$

$$\begin{aligned}
\text{Beliefs:}\quad b_r(\mathbf{x}_r) &= \beta_r(\mathbf{x}_r)\, m_{su}(\mathbf{x}_u), & b_t(\mathbf{x}_t) &= \beta_t(\mathbf{x}_t)\, m_{sv}(\mathbf{x}_v)\, m_{uw}(\mathbf{x}_w) \\
b_s(\mathbf{x}_s) &= \beta_s(\mathbf{x}_s)\, m_{ru}(\mathbf{x}_u)\, m_{tv}(\mathbf{x}_v), & b_u(\mathbf{x}_u) &= \beta_u(\mathbf{x}_u)\, m_{ru}(\mathbf{x}_u)\, m_{su}(\mathbf{x}_u) \\
b_w(\mathbf{x}_w) &= \beta_w(\mathbf{x}_w)\, m_{uw}(\mathbf{x}_w), & b_v(\mathbf{x}_v) &= \beta_v(\mathbf{x}_v)\, m_{sv}(\mathbf{x}_v)\, m_{tv}(\mathbf{x}_v)\, m_{uw}(\mathbf{x}_w)
\end{aligned}$$

Notice that the above-mentioned loop is now broken, since m_{vw} does not exist anymore. This means that the messages in the above algorithm will converge after just one round of updates (performed in the correct order). This of course is not surprising; based on the discussion above, this algorithm is no more than the belief propagation algorithm on the junction tree of Figure 6(b). □

6 Experimental Results

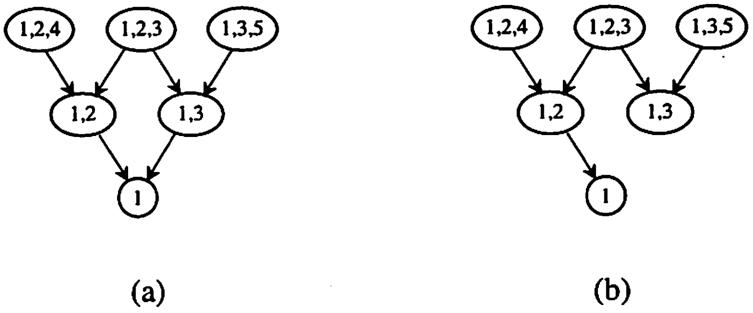
In the previous section we proved that the fixed points of GBP algorithms on any graphical representation for a poset R coincide with the solutions to the Kikuchi approximation problem of Section 3.2. We further argued that the algorithm on the minimal graph S_R has the smallest complexity per each iteration. Two important questions are not addressed in this paper so far: 1) *how close are the Kikuchi approximations to the true marginals?* And 2) *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 simulation results.

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 potential functions $\{\alpha_r(\mathbf{x}_r)\}$, 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 which are updated before their requisite set of edges have 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 (31). The value of w is always between 0 and 1, with $w = 1$ corresponding to (31). 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, i.e. the number of additions, multiplications and divisions involved in update rules of (31), 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 for each algorithm. The results are reported below:

Poset 1: The Hasse diagram of this poset has one loop, but the minimal graph is loop-free. 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 iteration (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 iteration by iteration and has less complexity per iteration.

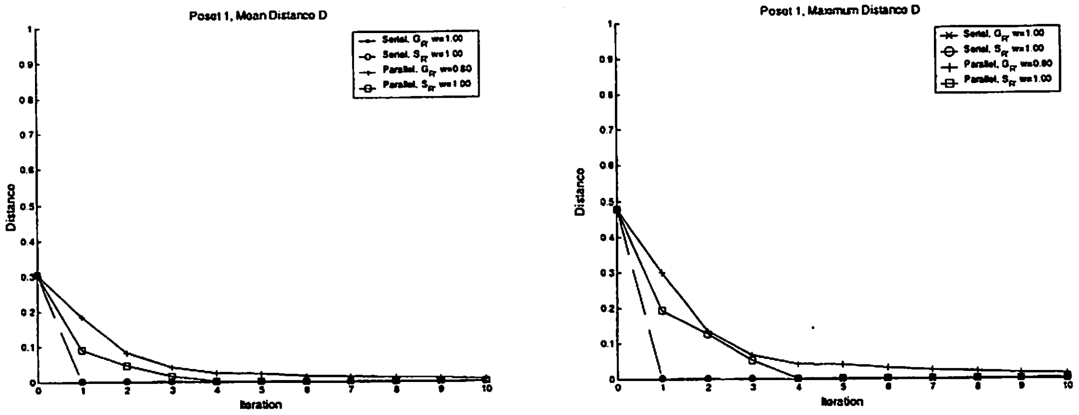


(a)

(b)

Figure 7: Posets 1. (a) The Hasse graph G_R, and (b) the minimal graph S_R

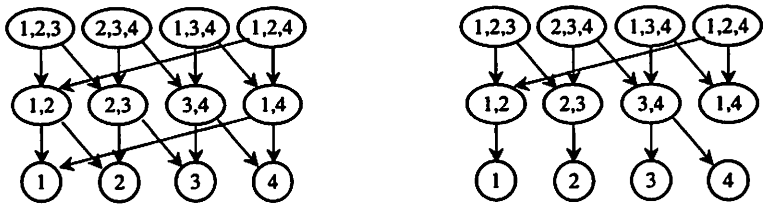
Poset 2: The Hasse diagram of this poset has five loops. All but one of these loops are broken in the minimal graph. There is a saving of 46.2% per each iteration of GBP on the minimal graph compared to that on the Hasse graph.



(a) (b)

Figure 8: Simulation Results on Poset 1.

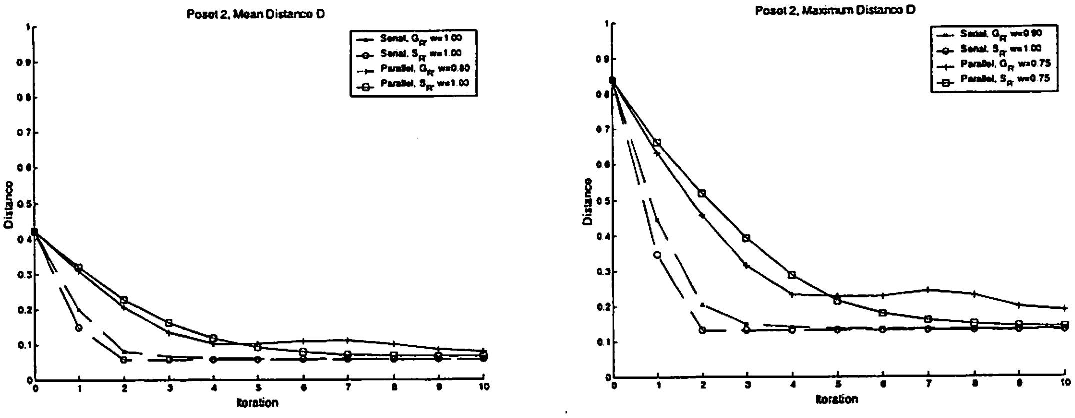
The Kikuchi approximations are at an average distance of about 0.05 from the true marginals, while the worst estimates have distance of about 0.13. Again the serial algorithms converge very quickly, although the one on the Hasse graph requires a slight damping. Comparing the parallel algorithms, the one on the minimal graph clearly outperforms the one on the full Hasse graph, even with equal damping factors.



(a) (b)

Figure 9: Posets 2. (a) The Hasse graph G_R, and (b) the minimal graph S_R

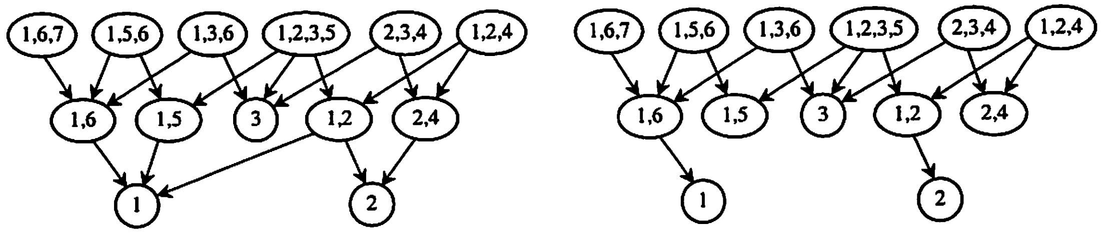
Poset 3: The Hasse diagram of this poset has five loops, whereas the minimal graph has only two loops. There is a saving of 28.5% per each iteration of GBP on the minimal graph compared to that on the Hasse graph. The Kikuchi approximations are at an average distance of about 0.05 from the true marginals, 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



(a) (b)

Figure 10: Simulation Results on Poset 2.

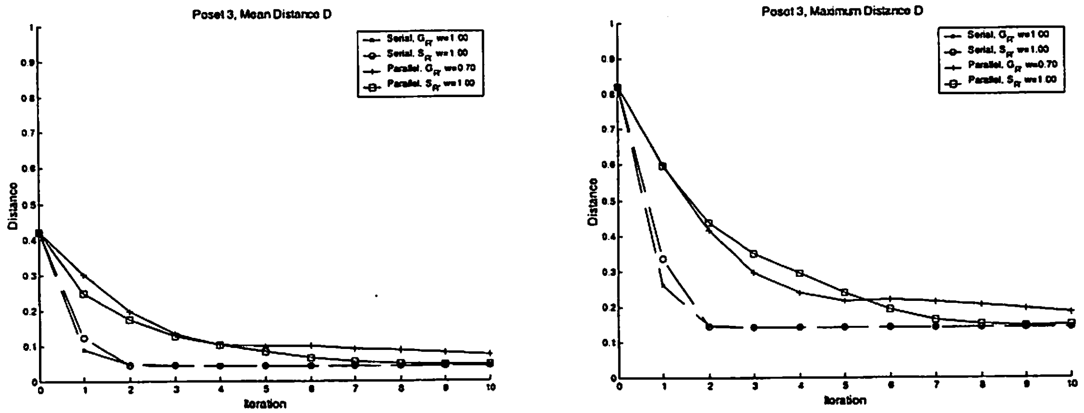
the minimal graph again outperforms that on the full Hasse graph, the latter requiring a damping factor $w = 0.70$ to avoid oscillations.



(a) (b)

Figure 11: Posets 3. (a) The Hasse graph G_R, and (b) the minimal graph S_R

At least for the simple posets considered here, the less complex GBP algorithm on the minimal graph, developed in this paper, 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 suggests that there is considerable saving in the complexity to be gained by using the algorithm on the minimal graph.



(a) (b)

Figure 12: Simulation Results on Poset 3.

7 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. and McEliece, R. (2000). The generalized distributive law. *IEEE Trans. Inform. Theory*, 46(2):325–343.

Aji, S. and McEliece, R. (2001). The generalized distributive law and free energy minimization. In *Proceedings of the Allerton Conference on Communication, Control, and Computing*, pages 672–681.

Berrou, C., Glavieux, A., and Thitimajshima, P. (1993). Near shannon limit error-correcting coding and decoding: Turbo-codes. In *Proceedings of the IEEE International Conference on Communications*, number 2, pages 1064–1070, Geneva.

Cormen, T., Leiserson, C., Rivest, R., and Stein, C. (2001). *Introduction to Algorithms*. McGraw-Hill, Cambridge, MA.

Cover, T. and Thomas, J. (1991). *Elements of Information Theory.* Wiley-Interscience, New York, NY.

Cowell, R., Dawid, A., Lauritzen, S., and Spiegelhalter, D. (1999). *Probabilistic Networks and Expert Systems.* Springer-Verlag, New York, NY.

Divsalar, D., Jin, H., and McEliece, R. (1998). Coding theorems for 'turbo-like' codes. In *Proceedings of the Allerton Conference on Communication, Control, and Computing*, pages 201–210.

Federer, H. (1969). *Geometric Measure Theory.* Springer-Verlag, New York, NY.

Gallager, R. (1963). *Low-Density Parity-Check Codes.* MIT Press, Cambridge, MA.

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. and Kroemer, H. (1980). *Thermal Physics.* New York, NY.

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

MacKay, D. and Neal, R. (1995). Good codes based on very sparse matrices. In *Cryptography and Coding: 5th IMA Conference*, number 1025, pages 100–111. Springer-Verlag, Berlin.

McEliece, R., MacKay, D., and 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. and Yildrim, M. (2003). Belief propagation on partially ordered sets. In *Mathematical Systems Theory in Biology, Communications, Computation, and Finance*, pages 275–300. Springer.

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

Pakzad, P. and Anantharam, V. Belief propagation and statistical physics. In *Conference on Information Sciences and Systems (CISS 2002)*, Princeton University.

Pakzad, P. and Anantharam, V. (2004). A new look at the generalized distributive law. To appear in IEEE Trans. Inform. Theory.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA.

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

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

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

Stanley, R. (1986). *Enumerative Combinatorics*, volume I. Wadsworth & Brooks/Cole, Monterey, CA.

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

Wainwright, M. and Jordan, M. (2003). Graphical models, exponential families, and variational inference. Technical Report 649, U.C. Berkeley, Dept. of Statistics.

Walrand, J. and Varaiya, P. (1996). *High-Performance Communication Networks*. Morgan Kaufmann, San Francisco, CA.

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

Welling, M. and 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*, pages 554–561.

Wicker, S. (1995). *Error Control Systems for Digital Communication and Storage.* Prentice Hall, Upper Saddle River, NJ.

Yedidia, J., Freeman, W., and Weiss, Y. (2001). Bethe free energy, Kikuchi approximations, and belief propagation algorithms. Technical Report TR2001-16, Mitsubishi Electronic Research Lab.

Yedidia, J., Freeman, W., and Weiss, Y. (2002). Constructing free energy approximations and generalized belief propagation algorithms. Technical Report TR2002-35, Mitsubishi Electronic 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.