6.962 Week 6 Summary: Approximate Inference Techniques for Graphs with Cycles

Erik Sudderth

March 15, 2001

1 Introduction

Graphical models are a tool for compactly specifying the probabilistic relationships between a set of random variables. They are used in application areas as diverse as coding, image processing, and artificial intelligence. For many practical graphs, exact statistical inference is too complex to be tractable. This has motivated the development of a wide variety of approximate inference techniques.

Recently, there has been a huge amount of interest in iterative estimation algorithms. This has been primarily motivated by the amazing empirical performance of a class of error–correcting codes known as turbo codes. The turbo decoding algorithm has been shown to be equivalent to Pearl's belief propagation (BP) algorithm [4, 6] when applied to a graph with cycles. However, BP was only "supposed" to work for acyclic graphs. This has led to a variety of studies which attempt to understand the behavior of BP on graphs with cycles [7–9].

This summary reviews some recent ideas which provide further insight into the specific approximation employed by loopy BP, and make the connections between BP and other approximate inference techniques like mean field theory more apparent. The material is primarily drawn from two tutorial papers by Yedidia [11] and Jordan et. al. [2]. The focus will be on the structural similarities which connect the various algorithms.

2 Graphical Model Fundamentals

Graphical models provide a powerful general framework for encoding the statistical structure of a set of random variables. In this section, we briefly review the means by which graphs encode conditional independencies. We then describe how graphs may be used to solve statistical inference problems, and motivate the need for approximate inference techniques.

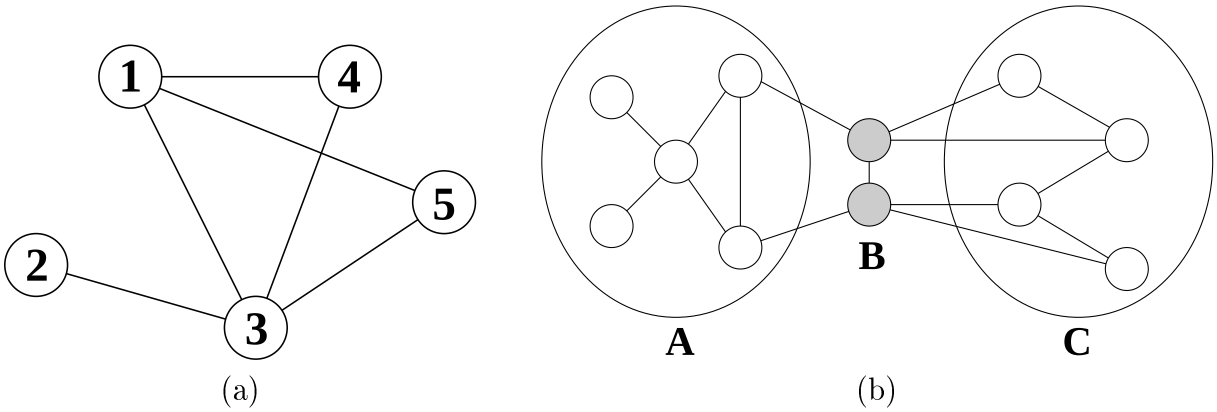

Figure 1: (a) An undirected graphical model representing five random variables. (b) The random variables in sets A and C are conditionally independent given set B.

2.1 Undirected Graphs and Conditional Independence

A graph $\mathcal{G}$ is defined by a set of nodes $\mathcal{S}$ and edges $\mathcal{E}$. Each node $s_i \in \mathcal{S}$ represents a random variable x_i. Let $N \triangleq |\mathcal{S}|$ be the number of nodes in the graph. Each edge $(i, j) \in \mathcal{E}$ connects two nodes s_i and s_j. We consider only undirected graphs, meaning that the edge (i, j) may be equivalently represented as (j, i). Figure 1(a) shows a typical undirected graphical model.

Taken together, the set of edges $\mathcal{E}$ implicitly defines a set of conditional independencies among the random variables. Let $\mathcal{X} = \bigcup_{s_i \in \mathcal{S}} x_i$. Then, if we define the neighborhood of x_i as $\mathcal{N}(x_i) = \{x_j \mid (i, j) \in \mathcal{E}\}$, we have

$$p\left(x_i \mid \mathcal{X} \setminus x_i\right) = p\left(x_i \mid \mathcal{N}(x_i)\right) \tag{1}$$

Conditioned on its immediate neighbors, the probability distribution of a given node is independent of the rest of the graph. A few of the many conditional independencies implied by the graphs in Figure 1 are

$$\begin{aligned} p\left(x_5 \mid x_1, x_2, x_3, x_4\right) &= p\left(x_5 \mid x_1, x_3\right) \\ p\left(x_A \mid x_B, x_C\right) &= p\left(x_A \mid x_B\right) \end{aligned}$$

Alternatively, conditioned on a given set of nodes, the probability distributions of disjoint subsets of the graph separated by those nodes are independent. For example, in Figure 1, this implies that

$$\begin{aligned} p\left(x_2, x_4, x_5 \mid x_1, x_3\right) &= p\left(x_2 \mid x_1, x_3\right) p\left(x_4 \mid x_1, x_3\right) p\left(x_5 \mid x_1, x_3\right) \\ p\left(x_A, x_C \mid x_B\right) &= p\left(x_A \mid x_B\right) p\left(x_C \mid x_B\right) \end{aligned}$$

Random variables which satisfy equation (1) are said to be Markov with respect to the undirected graph $\mathcal{G}$. The resulting model is also known as a Markov random field (MRF).

When working with undirected graphs, certain sets of nodes known as cliques are particularly important. A clique is defined to be a set of nodes in which every node is *directly* connected to every other node in the clique. For example, in Figure 1(a) the sets $\{x_1, x_3\}$,

$\{x_1, x_3, x_4\}$, and $\{x_1, x_3, x_5\}$ are all cliques. However, $\{x_1, x_3, x_4, x_5\}$ is *not* a clique because there is no edge between x_4 and x_5. A maximal clique is a clique which is not a strict subset of any other clique. For example, in Figure 1(a) $\{x_1, x_3, x_4\}$ is a maximal clique, but $\{x_1, x_3\}$ is not maximal.

When dealing with graphical models, an important issue is to determine which of the many possible distributions on $p(\mathcal{X})$ satisfy the conditional independencies implied by a given graph $\mathcal{G}$. This question is answered by the Hammersley–Clifford Theorem [1, 3].

Theorem 1 (Hammersley–Clifford) *Let $\mathcal{G}$ be a given undirected graph defined on a set of random variables $\mathcal{X}$, and let $\mathcal{C}$ be the set of all maximal cliques of $\mathcal{G}$. Then a positive distribution $p(\mathcal{X})$ satisfies the conditional independencies implied by $\mathcal{G}$ if and only if it can be written in the factorized form*

$$p(\mathcal{X}) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C) \tag{2}$$

$$Z = \sum_{\mathcal{X}} \prod_{C \in \mathcal{C}} \psi_C(x_C) \tag{3}$$

where $\psi_C(x_C)$ is an arbitrary positive function defined over the elements of the maximal clique C, and Z is a normalization constant.

Using terminology drawn from statistical mechanics, we often refer to the positive clique functions $\psi_C(x_C)$ as potentials and the normalizing constant Z as the partition function. The resulting distribution is sometimes called a Gibbs distribution. Due to the positivity of the clique potentials, it is possible to define a set of log potential functions $\phi_C(x_C) \triangleq -\log \psi_C(x_C)$ and rewrite $p(\mathcal{X})$ as a Boltzmann distribution

$$p(\mathcal{X}) = \frac{1}{Z} \exp\left\{ -\sum_{C \in \mathcal{C}} \phi_C(x_C) \right\} \tag{4}$$

Applying Theorem 1 to the graph in Figure 1(a), we find that the joint density must factorize in the following form:

$$p(x_1, x_2, x_3, x_4, x_5) = \frac{1}{Z} \psi_{2,3}(x_2, x_3)\, \psi_{1,3,4}(x_1, x_3, x_4)\, \psi_{1,3,5}(x_1, x_3, x_5)$$

Note, however, that it is possible to further decompose the maximal clique potentials. Often, it is convenient to make all of the cliques pairwise, which for Figure 1(a) would provide a decomposition of the form

$$p(\mathcal{X}) = \frac{1}{Z} \psi_{2,3}(x_2, x_3)\, \psi_{1,3}(x_1, x_3)\, \psi_{1,4}(x_1, x_4)\, \psi_{1,5}(x_1, x_5)\, \psi_{3,4}(x_3, x_4)\, \psi_{3,5}(x_3, x_5)$$

For convenience, in the remainder of this summary we will assume that all clique potentials are functions of either one or two nodes, allowing the distribution $p(\mathcal{X})$ to be written as

$$p(\mathcal{X}) = \frac{1}{Z} \prod_{(i,j) \in \mathcal{E}} \psi_{i,j}(x_i, x_j) \prod_{s_i \in \mathcal{S}} \psi_i(x_i) \tag{5}$$

$$= \frac{1}{Z} \exp\left\{ -\sum_{(i,j) \in \mathcal{E}} \phi_{i,j}(x_i, x_j) - \sum_{s_i \in \mathcal{S}} \phi_i(x_i) \right\} \tag{6}$$

where $\phi_{i,j}(x_i, x_j) \triangleq -\log \psi_{i,j}(x_i, x_j)$ and $\phi_i(x_i) \triangleq -\log \psi_i(x_i)$.

2.2 Inference using Graphical Models

In a Bayesian modeling framework, we use a graph to define a structured prior $p(\mathcal{X})$ as in equation (5). We then associate a noisy measurement $y_i \in \mathcal{Y}$ with each node $s_i \in \mathcal{S}$. We assume that each observation y_i is related to the "hidden" node x_i, which we do not observe, by a given conditional distribution

$$\begin{aligned} p(y_i \mid \mathcal{X}) &= p(y_i \mid x_i) \triangleq \rho_i(y_i; x_i) && (7) \\ p(\mathcal{Y} \mid \mathcal{X}) &= \prod_{s_i \in \mathcal{S}} \rho_i(y_i; x_i) && (8) \end{aligned}$$

Given a set of observations $\mathcal{Y}$, we would like to compute the conditional distribution

$$\begin{aligned} p(\mathcal{X} \mid \mathcal{Y}) &= \frac{p(\mathcal{X})\, p(\mathcal{Y} \mid \mathcal{X})}{p(\mathcal{Y})} \\ &= \frac{1}{Z(\mathcal{Y})} \prod_{(i,j) \in \mathcal{E}} \psi_{i,j}(x_i, x_j) \prod_{s_i \in \mathcal{S}} \psi_i(x_i)\, \rho_i(y_i; x_i) && (9) \end{aligned}$$

Comparing equation (9) with equation (5), we see that the conditional distribution $p(\mathcal{X} \mid \mathcal{Y})$ has the same graphical structure as the prior distribution $p(\mathcal{X})$. Because the observations $\mathcal{Y}$ are local, their only effect is to produce modified single-node clique potentials $\widetilde{\psi}_i(x_i) \triangleq \psi_i(x_i)\, \rho_i(y_i; x_i)$, which in turn produce a observation-dependent partition function $Z(\mathcal{Y})$.

For the remainder of this summary, we will assume that the node variables $\mathcal{X}$ take values from a discrete, finite set of dimension M. For such variables, statistical inference problems usually take one of two forms. In some cases, we are interested in the MAP estimate

$$\begin{aligned} \widehat{\mathcal{X}}_{\text{MAP}} &= \arg\max_{\mathcal{X}} p(\mathcal{X} \mid \mathcal{Y}) \\ &= \arg\max_{\mathcal{X}} \prod_{(i,j) \in \mathcal{E}} \psi_{i,j}(x_i, x_j) \prod_{s_i \in \mathcal{S}} \psi_i(x_i)\, \rho_i(y_i; x_i) && (10) \end{aligned}$$

In other situations, we would like to compute the conditional marginal distributions

$$\begin{aligned} p(x_i \mid \mathcal{Y}) &= \sum_{\mathcal{X} \setminus x_i} p(\mathcal{X} \mid \mathcal{Y}) \\ &= \frac{1}{Z(\mathcal{Y})} \sum_{\mathcal{X} \setminus x_i} \prod_{(i,j) \in \mathcal{E}} \psi_{i,j}(x_i, x_j) \prod_{s_i \in \mathcal{S}} \psi_i(x_i)\, \rho_i(y_i; x_i) && (11) \end{aligned}$$

Consider the direct evaluation of each of these estimates. To determine $\widehat{\mathcal{X}}_{\text{MAP}}$, we must evaluate all M^N possible configurations of $\mathcal{X}$, where N is the number of nodes and M is the number of possible values assumed by each node. To determine $p(x_i \mid \mathcal{Y})$, we must evaluate the partition function $Z(\mathcal{Y})$ which requires summing over M^N terms. In many applications such as image processing, it is common to have $N \approx 10^5$, in which case these calculations become completely intractable. This motivates the development of approximate inference algorithms.

For the remainder of this summary, we will focus on the problem of computing conditional marginals $p(x_i \mid \mathcal{Y})$. Note, however, that in many cases similar algorithms exist for MAP estimation. Also, given the graphical equivalence of $p(\mathcal{X} \mid \mathcal{Y})$ and $p(\mathcal{X})$, we will often focus simply on the problem of approximating the unconditional marginals $p(x_i)$, with the understanding that the same procedures can be used to approximate $p(x_i \mid \mathcal{Y})$.

3 Exact Inference for Tree–Structured Graphs

When a graph is tree–structured, efficient exact algorithms [6] exist for the computation of the marginal distributions $p(x_i \mid \mathcal{Y})$. These procedures are essentially extensions of dynamic programming techniques originally developed for inference on Markov chains. As these algorithms have a variety of important connections to approximation techniques for graphs with cycles, we briefly discuss them here.

3.1 Belief Propagation

By repeated application of Bayes' rule and the Markov properties of the graph, it is straightforward to show that any tree distribution may be factorized as

$$p(\mathcal{X}) = \prod_{(i,j)\in\mathcal{E}} \frac{p(x_i, x_j)}{p(x_i)\, p(x_j)} \prod_{s_i\in\mathcal{S}} p(x_i) \tag{12}$$

Here, $p(x_i, x_j)$ and $p(x_i)$ are the *exact* marginal distributions, and the partition function is simply $Z = 1$. Paralleling the discussion in §2.2, the conditional distribution $p(\mathcal{X} \mid \mathcal{Y})$ is simply given by

$$\begin{aligned} p(\mathcal{X} \mid \mathcal{Y}) &= \frac{1}{p(\mathcal{Y})} \prod_{(i,j)\in\mathcal{E}} \frac{p(x_i, x_j)}{p(x_i)\, p(x_j)} \prod_{s_i\in\mathcal{S}} p(x_i)\, p(y_i \mid x_i) \\ &\triangleq \frac{1}{p(\mathcal{Y})} \prod_{(i,j)\in\mathcal{E}} \psi_{i,j}(x_i, x_j) \prod_{s_i\in\mathcal{S}} \psi_i(x_i) \end{aligned} \tag{13}$$

Interestingly, for tree–structured priors $p(\mathcal{X})$ in the standard form of equation (12), the partition function of the conditional distribution $p(\mathcal{X} \mid \mathcal{Y})$ is equal to the likelihood $p(\mathcal{Y})$ of the observations.

We would like to compute $p(x_i \mid \mathcal{Y})$ for all $s_i \in \mathcal{S}$. For two neighboring nodes s_i and s_j, let $\mathcal{Y}_{j\setminus i}$ denote the set of all observations in the subtree rooted at s_j, excluding s_i and its descendants. We then have

$$p(x_i \mid \mathcal{Y}) = \frac{p(x_i)\, p(\mathcal{Y} \mid x_i)}{p(\mathcal{Y})} = \frac{p(x_i)\, p(y_i \mid x_i)}{p(\mathcal{Y})} \prod_{s_j\in\mathcal{N}(s_i)} p\left(\mathcal{Y}_{j\setminus i} \mid x_i\right) \tag{14}$$

We may now derive recursions for the conditional likelihoods $p\left(\mathcal{Y}_{j\setminus i} \mid x_i\right)$ by repeated application of Bayes' rule and the Markov properties of the graph. Doing so produces

$$p\left(\mathcal{Y}_{j\setminus i} \mid x_i\right) = \sum_{x_j} \frac{p\left(x_i, x_j\right) p\left(y_j \mid x_j\right)}{p\left(x_i\right)} \prod_{s_k \in \mathcal{N}(s_j)\setminus s_i} p\left(\mathcal{Y}_{k\setminus j} \mid x_k\right) \tag{15}$$

Suppose we associate the conditional likelihood $p\left(\mathcal{Y}_{j\setminus i} \mid x_i\right)$ with a "message" $m_{j\to i}\left(x_i\right)$ which s_j sends to s_i. Then, by substituting terms from equation (13) in equations (14) and (15), we arrive at the following equations:

$$p\left(x_i \mid \mathcal{Y}\right) = \alpha\psi_i\left(x_i\right) \prod_{s_j \in \mathcal{N}(s_i)} m_{j\to i}\left(x_i\right) \tag{16}$$

$$m_{j\to i}\left(x_i\right) = \sum_{x_j} \psi_{i,j}\left(x_i, x_j\right) \psi_j\left(x_j\right) \prod_{s_k \in \mathcal{N}(s_j)\setminus s_i} m_{k\to j}\left(x_j\right) \tag{17}$$

Here, α denotes a normalizing constant chosen so that $\sum_{x_i} p\left(x_i \mid \mathcal{Y}\right) = 1$.

Equations (16) and (17) have an appealing structure. Equation (16) expresses the marginal probability at a particular node as a product of a local potential term with the messages received from that node's immediate neighbors. Equation (17) expresses the outgoing messages at a particular node as a product of local potentials and incoming messages from the immediate neighbors, averaged over all possible values of that node. Thus, we can immediately formulate an algorithm where nodes iteratively receive a set of messages from their immediate neighbors, do some local computations, and send back a new set of messages to their neighbors. Each message is simply an M–dimensional vector of positive numbers, where M is the number of possible values assumed by each random variable x_i.

The resulting algorithm is known as "belief propagation" (BP). It is equivalent to the sum–product algorithm developed by the coding community. Careful examination of the message–passing equations shows that for tree–structured graphs, the resulting messages can produce the exact conditional marginals in only $\mathcal{O}(M^2N)$ operations, a huge savings over the direct cost of $\mathcal{O}(M^N)$. For trees, the messages have a nice probabilistic interpretation as $m_{i\to j}\left(x_j\right) = p\left(\mathcal{Y}_{i\setminus j} \mid x_j\right)$. Node s_i sends node s_j the likelihood of the data in its own subtree conditioned on each possible value of x_i. Intuitively, it makes sense to associate these likelihoods with s_j's "belief" in x_i.

3.2 Extensions to Graphs with Cycles

When $\mathcal{G}$ has cycles, the conditional independencies used to derive equations (16) and (17) no longer hold. There are a variety of potential ways to extend message–passing algorithms to more general graphs. One exact method is to cluster nodes in $\mathcal{G}$ until the reduced graph on the resulting "super–nodes" has no cycles. The BP algorithm could then be run on the clustered graph to produce exact marginal distributions. The process of clustering followed by an exact tree algorithm is known as the junction tree algorithm. Unfortunately, in order to maintain model consistency, large numbers of nodes from the original graph must be grouped together in the clustered graph, causing an exponential growth in computational cost. For most practical architectures, clustering does not produce a tractable exact inference algorithm.

An interesting approximate approach is simply to run the BP algorithm on the graph with cycles. In this case, the messages lose their strict interpretation in terms of probabilistic quantities. Intuitively, however, if the cycles in the graph are long, we expect the Markov properties employed in §3.1 to approximately hold, indicating that BP might produce reasonable approximations to the true marginals $p(x_i \mid \mathcal{Y})$. When there are many short loops, however, we would expect evidence to be overcounted, potentially producing "overconfidant" marginal distributions.

It turns out that this intuition is supported by numerical experiments [5]. In certain contexts, notably error–correcting codes, BP has proven to be remarkably effective. For other graphs, however, BP may produce extremely poor approximations or fail to converge at all. In the following sections, we will present some recent results which help clarify the specific approximation made in running BP on a graph with cycles.

4 Mean Field

In this section we present mean field theory as a method for approximating intractable Markov random fields [11]. Although mean field originally arose in the statistical physics literature, we motivate it from a more probabilistic perspective. The resulting algorithm is also related to a class of approximation techniques known as variational methods [2].

4.1 Approximating Distributions

One way to approach statistical inference problems on intractable graphs is to first approximate the complex model with a simpler structure. This amounts to approximating the distribution $p(\mathcal{X} \mid \mathcal{Y})$ by a distribution $q(\mathcal{X} \mid \mathcal{Y}, \lambda)$, where λ parameterizes a class of approximating distributions. The challenge is to determine a tractable means for finding the best λ.

One reasonable metric for choosing λ would be to minimize the Kullback–Leibler (KL) divergence $D(p \,||\, q)$:

$$\lambda^* = \arg\min_{\lambda} D(p(\mathcal{X} \mid \mathcal{Y}) \,||\, q(\mathcal{X} \mid \mathcal{Y}, \lambda)) \tag{18}$$

For example, it is straightforward to show that if we choose $q(\mathcal{X} \mid \mathcal{Y}, \lambda) = \prod_i q_i(x_i|\lambda_i)$ to be the fully factorized density, minimizing $D(p \,||\, q)$ will give $q_i(x_i|\lambda_i) = p(x_i \mid \mathcal{Y})$, exactly recovering the desired marginal distributions defined in equation (11). Unfortunately, $D(p \,||\, q)$ involves averages with respect to the *intractable* distribution $p(\mathcal{X} \mid \mathcal{Y})$, so this optimization will be no easier than the original problem.

Given that $D(p \,||\, q)$ is intractable, another natural approximation metric is $D(q \,||\, p)$. Since $D(q \,||\, p)$ involves averages with respect to the *tractable* distribution $q(\mathcal{X} \mid \mathcal{Y}, \lambda)$, we hope that it may lead to reasonable optimization problems. However, since the averages are with respect to the approximating distribution, it is not immediately clear why we should expect the minimization of $D(q \,||\, p)$ to produce good approximations.

To justify $D\left(q \,||\, p\right)$, consider the following lower bound on the likelihood $p\left(\mathcal{Y}\right)$ of the observations:

$$\begin{aligned}
\log p\left(\mathcal{Y}\right) &= \log \sum_{\mathcal{X}} p\left(\mathcal{X}, \mathcal{Y}\right) \\
&= \log \sum_{\mathcal{X}} q\left(\mathcal{X} \mid \mathcal{Y}, \lambda\right) \frac{p\left(\mathcal{X}, \mathcal{Y}\right)}{q\left(\mathcal{X} \mid \mathcal{Y}, \lambda\right)} \\
&\geq \sum_{\mathcal{X}} q\left(\mathcal{X} \mid \mathcal{Y}, \lambda\right) \log \left[\frac{p\left(\mathcal{X}, \mathcal{Y}\right)}{q\left(\mathcal{X} \mid \mathcal{Y}, \lambda\right)}\right]
\end{aligned} \tag{19}$$

Here, the third line follows directly from Jensen's inequality. The difference between the left and right sides of equation (19) is easily seen to be equal to the KL divergence $D\left(q \,||\, p\right)$. Therefore, by choosing λ to minimize $D\left(q \,||\, p\right)$, we pick the best possible lower bound on $p\left(\mathcal{Y}\right)$ in the class of approximating distributions $q\left(\mathcal{X} \mid \mathcal{Y}, \lambda\right)$. Thus, we will select the best approximating distribution $q\left(\mathcal{X} \mid \mathcal{Y}, \lambda^*\right)$ according to

$$\begin{aligned}
\lambda^* &= \arg\min_{\lambda} D\left(q\left(\mathcal{X} \mid \mathcal{Y}, \lambda\right) \,||\, p\left(\mathcal{X} \mid \mathcal{Y}\right)\right) \\
&= \arg\min_{\lambda} \sum_{\mathcal{X}} q\left(\mathcal{X} \mid \mathcal{Y}, \lambda\right) \log \frac{q\left(\mathcal{X} \mid \mathcal{Y}, \lambda\right)}{p\left(\mathcal{X} \mid \mathcal{Y}\right)}
\end{aligned} \tag{20}$$

4.2 Classical Mean Field Theory

We now return to the problem of approximating the marginal distributions $p\left(x_i\right)$ of a given joint distribution $p\left(\mathcal{X}\right)$ defined on a complex graph. In classical mean field theory, we choose the simplest possible approximating distribution by removing *all* of the edges in the graph. This corresponds to the fully factorized form $q\left(\mathcal{X}\right) = \prod_i q_i\left(x_i\right)$. Applying equation (20), we have

$$D\left(q \,||\, p\right) = \sum_{\mathcal{X}} \left[\prod_{s_i \in \mathcal{S}} q_i\left(x_i\right)\right] \log \left[\prod_{s_i \in \mathcal{S}} q_i\left(x_i\right)\right] - \sum_{\mathcal{X}} \left[\prod_{s_i \in \mathcal{S}} q_i\left(x_i\right)\right] \log p\left(\mathcal{X}\right) \tag{21}$$

Using the exponential representation of $p\left(\mathcal{X}\right)$ from equation (6) and simplifying, we may reduce this to

$$\begin{aligned}
D\left(q \,||\, p\right) &= \sum_{(i,j) \in \mathcal{E}} \sum_{x_i, x_j} q_i\left(x_i\right) q_j\left(x_j\right) \phi_{i,j}\left(x_i, x_j\right) + \sum_{s_i \in \mathcal{S}} \sum_{x_i} q_i\left(x_i\right) \phi_i\left(x_i\right) \\
&\quad + \sum_{s_i \in \mathcal{S}} \sum_{x_i} q_i\left(x_i\right) \log q_i\left(x_i\right)
\end{aligned} \tag{22}$$

We would like to choose the set of marginal distributions $q_i\left(x_i\right)$ which minimize $D\left(q \,||\, p\right)$. Note that even though we eventually want an independent distribution $q_i\left(x_i\right)$ at each node, the edges in $p\left(\mathcal{X}\right)$ cause terms proportional to $\phi_{i,j}\left(x_i, x_j\right)$ to appear in equation (22). These terms cause the optimization of the different marginals $q_i\left(x_i\right)$ to become coupled. It is this coupling during the optimization process which allows the fully factorized mean field approximation to capture some of the original graphical structure.

For nonhomogeneous fields, the simplest way to minimize $D(q \,||\, p)$ is to add a set of Lagrange multipliers γ_i enforcing the constraints that $\sum_{x_i} q_i(x_i) = 1$. Then, taking the derivative with respect to $q_i(x_i)$ and rearranging terms, we arrive at the following fixed point equation:

$$q_i(x_i) = \alpha \psi_i(x_i) \prod_{s_j \in \mathcal{N}(s_i)} \prod_{x_j} \psi_{i,j}(x_i, x_j)^{q_j(x_j)} \tag{23}$$

Here, α is a normalizing constant chosen so that $\sum_{x_i} q_i(x_i) = 1$. Comparing equation (23) to equations (16) and (17), we see that it is straightforward to define a message–passing algorithm for the optimization of the mean field distribution. However, in general the system of equations (23) may have many fixed points, and there is no guarantee that a local message–passing procedure will converge to the optimal factorized distribution.

4.3 Structured Mean Field and Variational Methods

The argument presented in §4.1 is a simple example of a variational method. In a variational method, we choose a family of tractable functions $g(x;\lambda)$ which provide bounds on some particular intractable function $f(x)$. We then attempt to find λ^* such that $g(x;\lambda^*)$ provides the tightest bound on $f(x)$. $g(x;\lambda^*)$ can then be employed as an approximation for $f(x)$ to facilitate otherwise intractable calculations. The variational parameters λ can be thought of as extra degrees of freedom which decouple dependencies in the original model at the expense of increasing the number of variables which must be dealt with.

By a clever choice of approximating bounds, a variety of techniques which improve the accuracy of the mean field approximation have been developed. One interesting class of methods goes by the name of structured mean field. These algorithms take advantage of the fact that we do not have to remove all of the edges from a graph to produce a tractable distribution. In particular, tractable inference procedures exist for tree–structured subgraphs. We can therefore formulate algorithms which iterate between calculating the variational parameters λ of the best subgraph, and then running an exact inference procedure as developed in §3.1.

By an appropriate choice of variational transformations, structured mean field techniques have been developed which show significant improvement over classical mean field approximations. Unfortunately, finding a set of approximating distributions which are both accurate and produce tractable optimization procedures can be difficult. Often, this choice must depend on the particular structure of the clique potentials comprising $p(\mathcal{X})$.

5 Understanding Loopy Belief Propagation

As discussed in §3, belief propagation (BP) is only exact for acyclic graphs. However, it is sometimes employed as an approximate inference technique on graphs with cycles. In this section, we provide an interpretation of the approximation that loopy BP represents using ideas drawn from statistical physics [11]. We then reinterpret this approximation from a probabilistic perspective, and motivate a class of generalized belief propagation (GBP) algorithms.

5.1 Gibbs Free Energy

In statistical physics, a quantity called the Gibbs free energy is often employed to develop approximations for intractable MRFs. Starting from equation (6), the Gibbs free energy G is given by

$$G = \sum_{\mathcal{X}} p(\mathcal{X}) \left[\sum_{(i,j)\in\mathcal{E}} \phi_{i,j}(x_i, x_j) + \sum_{s_i\in\mathcal{S}} \phi_i(x_i) \right] - \left[-\sum_{\mathcal{X}} p(\mathcal{X}) \log p(\mathcal{X}) \right] \tag{24}$$

G is written as a difference between two terms. The first term represents an average energy, while the second is the entropy $H(p(\mathcal{X}))$. For the purposes of this summary, G can simply be thought of as a quantity that, when minimized with respect to $p(\mathcal{X})$, produces the Boltzmann distribution of equation (6).

Exact minimization of G requires the calculation of the partition function Z, which is not feasible for MRFs with many cycles. This is often handled by assuming a particular parameterized form $q(\mathcal{X})$ for the true distribution $p(\mathcal{X})$. If $q(\mathcal{X})$ is chosen appropriately, the resulting approximate Gibbs free energy can be tractably minimized. For example, suppose that we choose $q(\mathcal{X}) = \prod_i q_i(x_i)$ to be the fully factorized distribution. The resulting approximate free energy is

$$G_{\mathrm{MF}} = \sum_{\mathcal{X}} \prod_i q_i(x_i) \left[\sum_{(j,k)\in\mathcal{E}} \phi_{j,k}(x_j, x_k) + \sum_{s_k\in\mathcal{S}} \phi_k(x_k) \right] + \sum_{\mathcal{X}} \prod_i q_i(x_i) \log \prod_i q_i(x_i) \tag{25}$$

Algebraic manipulation shows that equations (25) and (22) are equivalent. Therefore, we see that minimization of G_{MF} provides an alternate derivation of the mean field equations.

5.2 Bethe Free Energy

In order to improve on the mean field approximation, it is natural to consider more complicated approximate free energies. The Bethe free energy draws its inspiration from tree–structured models. By combining equations (24) and (12), we can show that the exact Gibbs free energy for a tree is given by

$$\begin{aligned} G_{\mathrm{B}} \;=\; & \sum_{(i,j)\in\mathcal{E}} \sum_{x_i,x_j} q_{i,j}(x_i, x_j) \left[\log \frac{q_{i,j}(x_i, x_j)}{q_i(x_i)\, q_j(x_j)} + \phi_{i,j}(x_i, x_j) \right] \\ & + \sum_{s_i\in\mathcal{S}} \sum_{x_i} q_i(x_i) \left[\log q_i(x_i) + \phi_i(x_i) \right] \end{aligned} \tag{26}$$

For graphs which are not tree–structured, it is not possible to exactly factorize $p(\mathcal{X})$ into products of nodewise marginals $p(x_i)$ and pairwise marginals $p(x_i, x_j)$ as in equation (12). The Bethe energy approximation simply uses the tree–structured free energy G_{B} even though the underlying graph is not a tree.

In order to minimize G_{B} with respect to $q_i(x_i)$ and $q_{i,j}(x_i, x_j)$, we first add Lagrange multipliers to enforce the various marginalization constraints:

$$
\begin{aligned}
\lambda_{i,j}(x_j) &\iff \sum_{x_i} q_{i,j}(x_i, x_j) = q_j(x_j) \\
\lambda_{j,i}(x_i) &\iff \sum_{x_j} q_{i,j}(x_i, x_j) = q_i(x_i) \\
\gamma_i &\iff \sum_{x_i} q_i(x_i) = 1 \\
\gamma_{ij} &\iff \sum_{x_i, x_j} q_{i,j}(x_i, x_j) = 1
\end{aligned}
$$

Taking the derivative of the resulting Lagrangian and manipulating, we arrive at the following two fixed–point equations:

$$
\begin{aligned}
q_i(x_i) &= \alpha \exp\left\{ \phi_i(x_i) + \frac{1}{|\mathcal{N}(s_i)| - 1} \sum_{s_j \in \mathcal{N}(s_i)} \lambda_{j,i}(x_i) \right\} &(27)\\
q_{i,j}(x_i, x_j) &= \alpha \exp\left\{ \phi_{i,j}(x_i, x_j) + \phi_i(x_i) + \phi_j(x_j) + \lambda_{i,j}(x_j) + \lambda_{j,i}(x_i) \right\} &(28)
\end{aligned}
$$

These equations are intriguingly similar to the BP equations derived in §3.1. In fact, by substituting the expressions for $q_i(x_i)$ and $q_{i,j}(x_i, x_j)$ back into the marginalization constraint equation, we find that they are *exactly* equivalent if we make the following association between Lagrange multipliers $\lambda_{i,j}(x_j)$ and messages $m_{i \to j}(x_j)$:

$$
\lambda_{i,j}(x_j) = \sum_{s_k \in \mathcal{N}(s_j) \setminus s_i} \log m_{k \to j}(x_j) \tag{29}
$$

We can therefore interpret the messages passed by the belief propagation algorithm as exponentiated Lagrange multipliers which enforce the pairwise marginalization constraints. These constraints arise naturally from the minimization of an approximate free energy G_{B} which is exact for tree–structured graphs. Unlike the probabilistic interpretation given in §3.1, this association of BP messages with local consistency conditions holds for arbitrary graphs.

In §4.2 and §5.1, we have shown that the mean field approximation can be interpreted either as the minimization of an approximate free energy G_{MF} or as the minimization of the KL divergence $D(q \,||\, p)$. It is natural to wonder whether there is a KL interpretation for the Bethe tree approximation. It turns out that the Bethe free energy is equivalent to an approximate KL distance, where the approximation arises from setting

$$
\log q(\mathcal{X}) \approx \sum_{(i,j) \in \mathcal{E}} \log \frac{q_{i,j}(x_i, x_j)}{q_i(x_i)\, q_j(x_j)} + \sum_{s_i \in \mathcal{S}} \log q_i(x_i) \tag{30}
$$

This approximation can be derived from a combinatorial expansion known as the Möbius inversion formula [3].

5.3 Kikuchi Approximations and Generalized Belief Propagation

Based on the interpretation of BP as an algorithm for minimizing the Bethe free energy, Yedidia, Freeman, and Weiss [10] have derived a new class of approximate inference algorithms they refer to as generalized belief propagation (GBP). The messages employed by BP essentially act to enforce all of the necessary marginalization constraints for pairwise potentials. This ensures that the beliefs at all pairs of nodes will be consistent. However, it does *not* guarantee that the beliefs at larger clusters of nodes will all be internally consistent.

GBP algorithms work by simply considering larger clusters of nodes. They pass more messages so that all of the nodes in each cluster are constrained to have consistent beliefs. As the clusters grow, more constraints are enforced and the approximation accuracy (hopefully) improves. Just as the BP messages can be associated with Lagrange multipliers for the minimization of the Bethe free energy, GBP messages are Lagrange multipliers which enforce a set of higher–order free energies known as Kikuchi approximations. Of course, computational cost grows with the cluster size because larger messages must be employed. Nevertheless, GBP algorithms provide a first step towards dealing with problems for which BP produces poor results.

References

[1] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Jour. Royal Stat. Soc. B*, 41:192–223, 1974.

[2] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37:183–233, 1999.

[3] S. L. Lauritzen. *Graphical models*. Oxford University Press, Oxford, 1996.

[4] R.J. McEliece, D.J.C. McKay, and J.F. Cheng. Turbo decoding as an instance of Pearl's belief propagation algorithm. *IEEE Jour. Sel. Communication*, 16(2):140–152, February 1998.

[5] K. P. Murphy, Y. Weiss, and M. I. Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Uncertainty in AI*, 1999.

[6] J. Pearl. *Probabilistic reasoning in intelligent systems*. Morgan Kaufman, San Mateo, 1988.

[7] P. Rusmevichientong and B. Van Roy. An analysis of turbo decoding with Gaussian densities. In *NIPS 12*, pages 575–581. MIT Press, 2000.

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

[9] Y. Weiss and W.T. Freeman. Correctness of belief propagation in Gaussian graphical models of arbitrary topology. Technical report, University California at Berkeley, 1999. Available at `http://www.cs.berkeley.edu/ yweiss/ncpaperweb.ps.gz`.

[10] J. Yedidia, W. T. Freeman, and Y. Weiss. Generalized belief propagation. In *NIPS 13*, page to appear. MIT Press, 2001.

[11] J. S. Yedidia. An idiosyncratic journey beyond mean field theory. Technical Report TR-2000-27, MERL, 2000.