Linear Response Algorithms for Approximate Inference in Graphical Models

Max Welling
Department of Computer Science
University of Toronto
10 King's College Road, Toronto M5S 3G4 Canada
welling@cs.toronto.edu

Yee Whye Teh
Computer Science Division
University of California at Berkeley
Berkeley CA94720 USA
ywteh@eecs.berkeley.edu

Keywords: Loopy Belief Propagation, Mean Field, Linear Response, Inference

Abstract

Belief propagation (BP) on cyclic graphs is an efficient algorithm for computing approximate marginal probability distributions over single nodes and neighboring nodes in the graph. It does however not prescribe a way to compute joint distributions over pairs of distant nodes in the graph. In this paper we propose two new algorithms for approximating these pairwise probabilities, based on the linear response theorem. The first is a propagation algorithm which is shown to converge if belief propagation converges to a stable fixed point. The second algorithm is based on matrix inversion. Applying these ideas to Gaussian random fields we derive a propagation algorithm for computing the inverse of a matrix.

1 Introduction

Like Markov chain Monte Carlo sampling and variational methods, belief propagation (BP) has become an important tool for approximate inference on graphs with cycles. Especially in the field of "error correction decoding", it has brought performance very close to the Shannon limit [1]. BP was studied in a number of papers which have gradually increased our understanding of the convergence properties and accuracy of the algorithm [15, 13]. In particular, recent developments show that the stable fixed points are local minima of the bethe free energy [17, 3]. This insight paved the way for more sophisticated "generalized belief propagation" algorithms [18] and convergent alternatives to BP [19, 9]. Other developments also include the "expectation propagation" algorithm designed to propagate sufficient statistics of members of the exponential family [6].

Despite its success, BP does not provide a prescription to compute joint probabilities over pairs of non-neighboring nodes in the graph. When the graph is a tree, there is a single chain connecting any two nodes, and dynamic programming can be used to efficiently integrate out the internal variables. However, when cycles exist, it is not clear what approximate procedure is appropriate. It is precisely this problem that we will address in this paper. We show that the required estimates can be obtained by computing the "sensitivity" of the node marginals to small changes in the node potentials. Based on this idea, we present two algorithms to estimate the joint probabilities of arbitrary pairs of nodes.

These results are interesting in the inference domain but may also have future applications to learning graphical models from data. For instance, information about dependencies between random variables is relevant for learning the structure of a graph and the parameters encoding the interactions. Another possible application area is "active learning". Since the node potentials encode the external evidence flowing into the network, and since we compute the sensitivity of the marginal distributions to changing this external evidence, one may use this information to search for good nodes to collect additional data for. For instance, nodes which have a big impact on the system seem good candidates.

The paper is organized as follows. Factor graphs are introduced in section 2. Section 3 reviews the Gibbs free energy and two popular approximations, namely the mean field and Bethe approximations. Then in section 4 we explain the ideas behind the linear response estimates of pairwise probabilities, and prove a number of useful properties that they satisfy. We derive an algorithm to compute the linear response estimates by propagating "super-messages" around the graph in section 5, while section 6 describes an alternative method based on inverting a matrix. Section 7 describes an application of linear response theory to Gaussian networks that gives a novel algorithm to invert matrices. In experiments (section 8) we compare the accuracy of the new estimates against other methods. Finally we conclude with a discussion of our work in section 9.

2 Factor Graphs

Let V index a collection of random variables $\{X_i\}_{i \in V}$. Let x_i denote values of X_i. For a subset of nodes $\alpha \subset V$ let $X_\alpha = \{X_i\}_{i \in \alpha}$ be the variable associated with that subset, and x_α be values of X_α. Let A be a family of such subsets of V. The probability distribution over $X \doteq X_V$ is assumed to have the following form,

$$P_X(X = x) = \frac{1}{Z} \prod_{\alpha \in A} \psi_\alpha(x_\alpha) \prod_{i \in V} \psi_i(x_i) \tag{1}$$

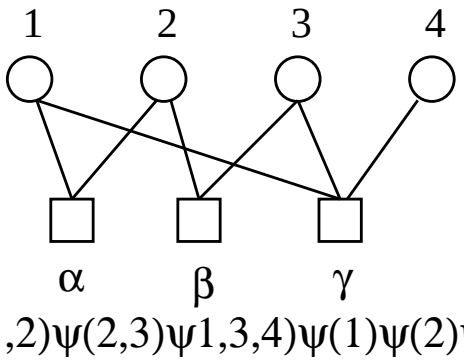

Figure 1: Example of a factor graph.

where ψ_α, ψ_i are positive potential functions defined on subsets and single nodes respectively. Z is the normalization constant (or partition function), given by

$$Z = \sum_x \prod_{\alpha \in A} \psi_\alpha(x_\alpha) \prod_{i \in V} \psi_i(x_i) \tag{2}$$

where the sum runs over all possible states x of X. In the following we will write $P(x) \doteq P_X(X = x)$ for notational simplicity. The decomposition of (1) is consistent with a factor graph with function nodes over X_α and variables nodes X_i. Figure 1 shows an example. Neighbors in a factor graph are defined as nodes that are connected by an edge (e.g. subset α and variable 2 are neighbors in figure 1). For each $i \in V$ denote its neighbors by $\mathcal{N}_i = \{\alpha \in A : \alpha \ni i\}$ and for each subset α its neighbors are simply $\mathcal{N}_\alpha = \{i \in V : i \in \alpha\}$.

Factor graphs are a convenient representation for structured probabilistic models and subsume undirected graphical models and acyclic directed graphical models [5]. Further, there is a simple message passing algorithm for approximate inference that generalizes the belief propagation algorithms on both undirected and acyclic directed graphical models. For that reason we will state the results of this paper in the language of factor graphs.

3 The Gibbs Free Energy

Let $B(x)$ be a *variational* probability distribution, and let b_α, b_i be its marginal distributions over $\alpha \in A$ and $i \in V$ respectively. Consider minimizing the following

objective, called the Gibbs free energy,

$$G(B) = -\sum_{\alpha}\sum_{x_\alpha} b_\alpha(x_\alpha)\log\psi_\alpha(x_\alpha) - \sum_{i}\sum_{x_i} b_i(x_i)\log\psi_i(x_i) - H(B) \quad (3)$$

where $H(B)$ is the entropy of $B(x)$,

$$H(B) = -\sum_{\mathbf{x}} B(x)\log B(x) \quad (4)$$

It is easy to show that the Gibbs free energy is precisely minimized at $B(x) = P(x)$. In the following we will use this variational formulation to describe two types of approximations: the mean field and the Bethe approximations.

3.1 The Mean Field Approximation

The mean field approximation uses a restricted set of variational distributions, namely those which assume independence between all variables x_i: $B^{\texttt{MF}}(x) \doteq \prod_i b_i^{\texttt{MF}}(x_i)$. Plugging this into the Gibbs free energy we get,

$$\begin{aligned} G^{\texttt{MF}}(\{b_i^{\texttt{MF}}\}) = \quad & -\sum_{\alpha}\sum_{x_\alpha}\left(\prod_{i\in\alpha} b_i^{\texttt{MF}}(x_i)\right)\log\psi_\alpha(x_\alpha) \\ & -\sum_{i}\sum_{x_i} b_i^{\texttt{MF}}(x_i)\log\psi_i(x_i) - H^{\texttt{MF}}(\{b_i^{\texttt{MF}}\}) \end{aligned} \quad (5)$$

where $H^{\texttt{MF}}$ is the mean field entropy

$$H^{\texttt{MF}}(\{b_i^{\texttt{MF}}\}) = -\sum_{i}\sum_{x_i} b_i^{\texttt{MF}}(x_i)\log b_i^{\texttt{MF}}(x_i) \quad (6)$$

Minimizing this with respect to $b_i^{\texttt{MF}}(x_i)$ (holding the remaining marginal distributions fixed) we derive the following update equation,

$$b_i^{\texttt{MF}}(x_i) \leftarrow \frac{1}{\gamma_i}\psi_i(x_i)\exp\left(\sum_{\alpha\in\mathcal{N}_i}\sum_{x_{\alpha\setminus i}}\log\psi_\alpha(x_\alpha)\prod_{j\in\mathcal{N}_\alpha\setminus i} b_j^{\texttt{MF}}(x_j)\right) \quad (7)$$

where γ_i is a normalization constant. Sequential updates which replace each $b_i^{\texttt{MF}}(x_i)$ by the RHS of (7) are a form of coordinate descent on the MF-Gibbs free energy which implies that they are guaranteed to converge to a local minimum.

3.2 The Bethe Approximation: Belief Propagation

The mean field approximation ignores all dependencies between the random variables and as such over-estimates the entropy of the model. To obtain a more accurate approximation we sum the entropies of the subsets $\alpha \in A$ and the nodes $i \in V$. However, this over-counts the entropies on the overlaps of the subsets $\alpha \in A$, which we therefore subtract off as follows,

$$H^{\mathtt{BP}}(\{b_\alpha^{\mathtt{BP}}, b_i^{\mathtt{BP}}\}) = -\sum_\alpha \sum_{x_\alpha} b_\alpha^{\mathtt{BP}}(x_\alpha) \log b_\alpha^{\mathtt{BP}}(x_\alpha) - \sum_i c_i \sum_{x_i} b_i^{\mathtt{BP}}(x_i) \log b_i^{\mathtt{BP}}(x_i) \tag{8}$$

where the over-counting numbers are $c_i = 1 - |\mathcal{N}_i|$. The resulting Gibbs free energy is thus given by [17],

$$\begin{aligned} G^{\mathtt{BP}}(\{b_i^{\mathtt{BP}}, b_\alpha^{\mathtt{BP}}\}) = &-\sum_\alpha \sum_{x_\alpha} b_\alpha^{\mathtt{BP}}(x_\alpha) \log \psi_\alpha(x_\alpha) \\ &-\sum_i \sum_{x_i} b_i^{\mathtt{BP}}(x_i) \log \psi_i(x_i) - H^{\mathtt{BP}}(\{b_i^{\mathtt{BP}}, b_\alpha^{\mathtt{BP}}\}) \end{aligned} \tag{9}$$

where the following local constraints need to be imposed,[1]

$$\sum_{x_{\alpha \setminus i}} b_\alpha^{\mathtt{BP}}(x_\alpha) = b_i^{\mathtt{BP}}(x_i) \qquad \forall \alpha \in A, i \in \alpha, x_i \tag{10}$$

in addition to the constraints that all marginal distributions should be normalized. It was shown in [17] that this constrained minimization problem may be solved by propagating messages over the links of the graph. Since the graph is bipartite we only need to introduce messages from factor nodes to variable nodes $m_{\alpha i}(x_i)$ and messages from variable nodes to factor nodes $n_{i\alpha}(x_i)$. The following fixed point equations can now be derived that solve for a local minimum of the BP-Gibbs free energy,

$$n_{i\alpha}(x_i) \leftarrow \psi_i(x_i) \prod_{\beta \in \mathcal{N}_i \setminus \alpha} m_{\beta i}(x_i) \tag{11}$$

$$m_{\alpha i}(x_i) \leftarrow \sum_{x_{\alpha \setminus i}} \psi_\alpha(x_\alpha) \prod_{j \in \mathcal{N}_\alpha \setminus i} n_{j\alpha}(x_j) \tag{12}$$

[1] Note that although the beliefs $\{b_i, b_\alpha\}$ satisfy local consistency constraints, they need not actually be globally consistent in that they do not necessarily correspond to the marginal distributions of a single probability distribution $B(x)$.

Finally, marginal distributions over factor nodes and variable nodes are expressed in terms of the messages as follows,

$$b_\alpha(x_\alpha) = \frac{1}{\gamma_\alpha}\psi_\alpha(x_\alpha)\prod_{i\in\mathcal{N}_\alpha} n_{i\alpha}(x_i) \tag{13}$$

$$b_i(x_i) = \frac{1}{\gamma_i}\psi_i(x_i)\prod_{\alpha\in\mathcal{N}_i} m_{\alpha i}(x_i) \tag{14}$$

where γ_i, γ_α are normalization constants.

On tree structured factor graphs there exists a scheduling such that each message needs to be updated only once in order to compute the exact marginal distributions on the factors and the nodes. On factor graphs with loops, iterating the messages do not always converge, but if they converge they often given accurate approximations to the exact marginals [7]. Further, the stable fixed points of the iterations can only be local minima of the BP-Gibbs free energy [3]. We note that theoretically there is no need to normalize the messages themselves (as long as one normalizes the estimates of the marginals), but that it is desired computationally to avoid numerical overflow or underflow.

4 Linear Response

The mean field and belief propagation algorithms described above provide estimates for single node marginals (both MF and BP) and factor node marginals (BP only), but not for joint marginal distributions of distant nodes. The linear response (LR) theory can be used to estimate joint marginal distributions over an arbitrary pair of nodes. For pairs of nodes inside a single factor, this procedure even improves upon the estimates that can be obtained from BP by marginalization of factor node marginals.

The idea here is to study changes in the system when we perturb the single node potentials,

$$\log\psi_i(x_i) = \log\psi_i^0(x_i) + \theta_i(x_i) \tag{15}$$

The superscript ^{0} indicates unperturbed quantities in the following. Let $\theta = \{\theta_i\}$ and define the free energy

$$F(\theta) = -\log\sum_x \prod_{\alpha\in A}\psi_\alpha(x_\alpha)\prod_{i\in V}\psi_i^0(x_i)e^{\theta_i(x_i)} \tag{16}$$

$-F(\theta)$ is the cumulant generating function for $P(X)$, up to irrelevant constants.

Differentiating $F(\theta)$ with respect to θ gives

$$-\left.\frac{\partial F(\theta)}{\partial \theta_j(x_j)}\right|_{\theta=0} = p_j(x_j) \tag{17}$$

$$-\left.\frac{\partial^2 F(\theta)}{\partial \theta_i(x_i)\partial \theta_j(x_j)}\right|_{\theta=0} = \left.\frac{\partial p_j(x_j)}{\partial \theta_i(x_i)}\right|_{\theta=0} = \begin{cases} p_{ij}(x_i,x_j) - p_i(x_i)p_j(x_j) & \text{if } i \neq j \\ p_i(x_i)\delta_{x_i,x_j} - p_i(x_i)p_j(x_j) & \text{if } i = j \end{cases} \tag{18}$$

where p_i, p_{ij} are single and pairwise marginals of $P(x)$. Hence second order perturbations in the system (18) gives the covariances between any two nodes of the system. The desired joint marginal distributions are then obtained by adding back the $p_i(x_i)p_j(x_j)$ term. Expressions for higher order cumulants can be derived by taking further derivatives of $-F(\theta)$.

4.1 Approximate Linear Response

Notice from (18) that the covariance estimates are obtained by studying the perturbations in $p_j(x_j)$ as we vary $\theta_i(x_i)$. This is not practical in general since calculating $p_j(x_j)$ itself is intractable. Instead, we consider perturbations of approximate marginal distributions $\{b_j\}$. In the following we will assume that $b_j(x_j;\theta)$ are the beliefs at a local minimum of the approximate Gibbs free energy under consideration (possibly subject to constraints).

In analogy to (18), let $C_{ij}(x_i,x_j) = \left.\frac{\partial b_j(x_j;\theta)}{\partial \theta_i(x_i)}\right|_{\theta=0}$ be the linear response estimated covariance, and define the linear response estimated joint pairwise marginal as

$$b_{ij}^{\mathtt{LR}}(x_i,x_j) = C_{ij}(x_i,x_j) + b_i^0(x_i)b_j^0(x_j) \tag{19}$$

where $b_i^0(x_i) \doteq b_i(x_i;\theta=0)$. We will show that $b_{ij}^{\mathtt{LR}}$ and C_{ij} satisfy a number of important properties of joint marginals and covariances.

First we show that $C_{ij}(x_i,x_j)$ can be interpreted as the Hessian of a well-behaved convex function. We focus here on the Bethe approximation (the mean field case is simpler). First, let $\mathcal{C}$ be the set of beliefs that satisfy the constraints (10) and normalization constraints. The approximate marginals $\{b_i^0\}$ along with the joint marginals $\{b_\alpha^0\}$ form a local minimum of the Bethe-Gibbs free energy (subject to $b^0 \doteq \{b_i^0, b_\alpha^0\} \in \mathcal{C}$). Assume that b^0 is a *strict* local minimum[2] of $G^{\mathtt{BP}}$. That is there is an open domain $\mathcal{D}$ containing b^0 such that $G^{\mathtt{BP}}(b^0) < G^{\mathtt{BP}}(b)$ for each $b \in \mathcal{D} \cap \mathcal{C} \backslash b^0$. Now we can define

$$G^*(\theta) = \inf_{b \in \mathcal{D}\cap\mathcal{C}} G^{\mathtt{BP}}(b) - \textstyle\sum_{i,x_i} b_i(x_i)\theta_i(x_i) \tag{20}$$

[2]The strict local minimality is in fact attained if we use loopy belief propagation [3].

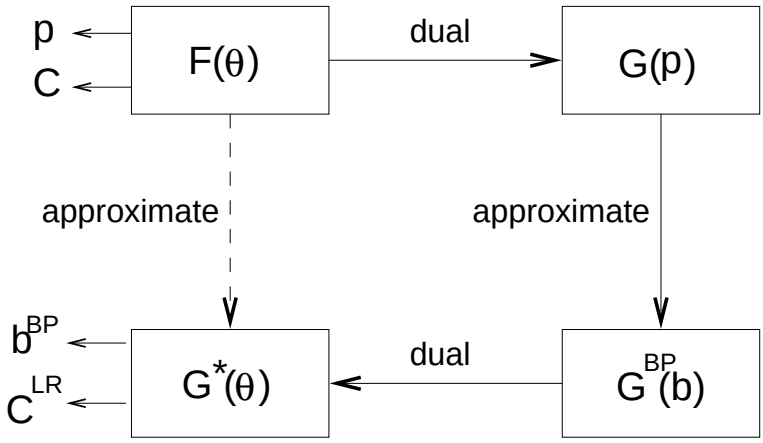

Figure 2: Diagrammatic representation of the different objective functions discussed in the paper. The free energy F is the cumulant generating function (up to a constant), G is the Gibbs free energy, G^{BP} is the Bethe-Gibbs free energy which is an approximation to the true Gibbs free energy and G^* is the approximate cumulant generating function. The actual approximation is performed in the dual space, while the dashed arrow indicates that the overall process gives G^* as an approximation to F.

$G^*(\theta)$ is a concave function since it is the infimum of a set of linear functions in θ. Further $G^*(0) = G(b^0)$ and since b^0 is a strict local minimum when $\theta = 0$, small perturbations in θ will result in small perturbations in b^0, so that G^* is well-behaved on an open neighborhood around $\theta = 0$. Differentiating $G^*(\theta)$, we get $\frac{\partial G^*(\theta)}{\partial \theta_j(x_j)} = -b_j(x_j;\theta)$ so that we now have

$$C_{ij}(x_i, x_j) = \left.\frac{\partial b_j(x_j;\theta)}{\partial \theta_i(x_i)}\right|_{\theta=0} = -\left.\frac{\partial^2 G^*(\theta)}{\partial \theta_i(x_i)\partial \theta_j(x_j)}\right|_{\theta=0} \tag{21}$$

In essence, we can interpret $G^*(\theta)$ as a *local* convex dual of $G^{\text{BP}}(b)$ (by restricting attention to $\mathcal{D}$). Since G^{BP} is an approximation to the exact Gibbs free energy [16], which is in turn dual to $F(\theta)$ [2], $G^*(\theta)$ can be seen as an approximation to $F(\theta)$ for small values of θ. For that reason we can take its second derivatives $C_{ij}(x_i, x_j)$ as approximations to the exact covariances (which are second derivatives of $-F(\theta)$). These relationships are shown pictorially in figure 2.

We now proceed to prove a number of important properties of the covariance C.

Theorem 1 *The approximate covariance satisfies the following symmetry:*

$$C_{ij}(x_i, x_j) = C_{ji}(x_j, x_i) \tag{22}$$

Proof: The covariances are second derivatives of $-G^*(\theta)$ at $\theta = 0$ and we can interchange the order of the derivatives since $G^*(\theta)$ is well-behaved on a neighborhood around $\theta = 0$. $\square$

Theorem 2 *The approximate covariance satisfies the following "marginalization" conditions for each x_i, x_j:*

$$\sum_{x'_i} C_{ij}(x'_i, x_j) = \sum_{x'_j} C_{ij}(x_i, x'_j) = 0 \tag{23}$$

As a result the approximate joint marginals satisfy local marginalization constraints:

$$\sum_{x'_i} b_{ij}^{\mathtt{LR}}(x'_i, x_j) = b_j^0(x_j) \qquad\qquad \sum_{x'_j} b_{ij}^{\mathtt{LR}}(x_i, x'_j) = b_i^0(x_i) \tag{24}$$

Proof: Using the definition of $C_{ij}(x_i, x_j)$ and marginalization constraints for b_j^0,

$$\sum_{x'_j} C_{ij}(x_i, x'_j) = \sum_{x'_j} \frac{\partial b_j(x'_j;\theta)}{\partial \theta_i(x_i)}\Big|_{\theta=0} = \frac{\partial}{\partial \theta_i(x_i)} \sum_{x'_j} b_j(x'_j;\theta)\Big|_{\theta=0} = \frac{\partial}{\partial \theta_i(x_i)} 1\Big|_{\theta=0} = 0 \tag{25}$$

The constraint $\sum_{x'_i} C_{ij}(x'_i, x_j) = 0$ follows from the symmetry (22), while the corresponding marginalization (24) follows from (23) and the definition of $b_{ij}^{\mathtt{LR}}$. $\square$

Since $-F(\theta)$ is convex, its Hessian matrix with entries given in (18) is positive semi-definite. Similarly, since the approximate covariances $C_{ij}(x_i, x_j)$ are second derivatives of a convex function $-G^*(\theta)$, we have:

Theorem 3 *The matrix formed from the approximate covariances $C_{ij}(x_i, x_j)$ by varying i and x_i over the rows and varying j, x_j over the columns is positive semi-definite.*

Using the above results we can reinterpret the linear response correction as a "projection" of the (only locally consistent) beliefs $\{b_i^0, b_\alpha^0\}$ onto a set of beliefs $\{b_i^0, b_{ij}^{\mathtt{LR}}\}$ that is both locally consistent (theorem 2) and satisfies the global constraint of being positive semi-definite (theorem 3). This is depicted in figure 3. Indeed the idea to include global constraints such as positive semi-definiteness in approximate inference algorithms was proposed in [11]. It is surprising that a simple post-hoc projection can achieve the same result.

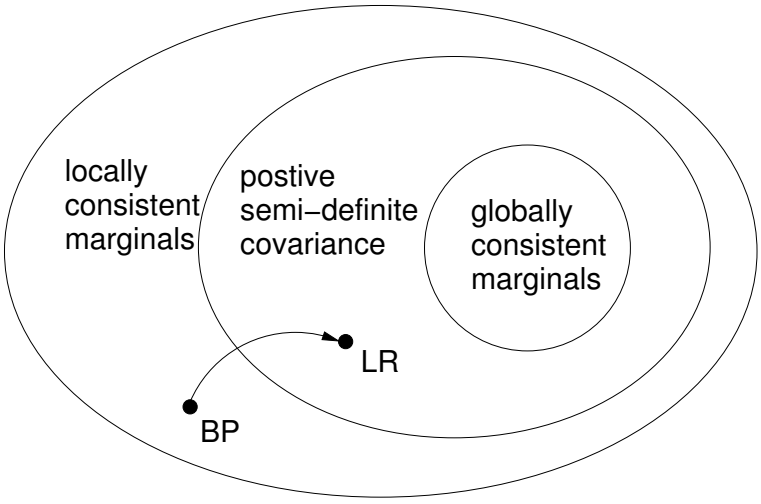

Figure 3: Various constraint sets discussed in the paper. The inner set is the set of all marginal distribution which are consistent with some global distribution $B(x)$, the outer set is the constraint set of all locally consistent marginal distributions, while the middle set consists of locally consistent marginal distributions with positive semi-definite covariance. The linear response algorithm performs a correction on the joint pairwise marginals such that the covariance matrix is symmetric and positive semi-definite, while all local consistency relations are still respected.

5 Propagation Algorithms for Linear Response

Although we have derived an expression for the covariance in the linear response approximation (21), we haven't yet explained how to efficiently compute it. In this section we derive a propagation algorithm to that end and prove some convergence results, while in the next section we will present an algorithm based on a matrix inverse.

Recall from (19) that we need the first derivative of $b_i(x_i;\theta)$ with respect to $\theta_j(x_j)$ at $\theta = 0$. This does not automatically imply that we need an analytic expression for $b_i(x_i;\theta)$ in terms of θ. Instead, we only need to keep track of first order dependencies by expanding all quantities and equations *up to first order* in θ. For the beliefs we write[3],

$$b_i(x_i;\theta) = b_i^0(x_i)\left(1 + \sum_{j,y_j} R_{ij}(x_i,y_j)\theta_j(y_j)\right) \tag{26}$$

[3]The unconventional form of this expansion will make subsequent derivations more transparent.

The "response matrix" $R_{ij}(x_i, y_j)$ measures the sensitivity of $\log b_i(x_i;\theta)$ at node i to a change in the log node potentials $\log \psi_j(y_j)$ at node j. Combining (26) with (21), we find that

$$C_{ij}(x_i, x_j) = b_i^0(x_i) R_{ij}(x_i, x_j) \tag{27}$$

The constraints (23) (which follow from the normalization of $b_i(x_i;\theta)$ and $b_i^0(x_i)$) translate into

$$\sum_{x_i} b_i^0(x_i) R_{ij}(x_i, y_j) = 0 \tag{28}$$

and it is not hard to verify that the following shift can be applied to accomplish this[4],

$$R_{ij}(x_i, y_j) \leftarrow R_{ij}(x_i, y_j) - \sum_{x_i} b_i^0(x_i) R_{ij}(x_i, y_j) \tag{29}$$

5.1 The Mean Field Approximation

Let us assume that we have found a local minimum of the MF-Gibbs free energy by iterating (7) until convergence. By inserting the expansions (15,26) into (7) and equating terms linear in θ we derive the following update equations for the response matrix in the MF approximation,

$$R_{ik}(x_i, y_k) \leftarrow \delta_{ik}\delta_{x_i y_k} + \sum_{\alpha \in \mathcal{N}_i} \sum_{x_{\alpha \setminus i}} \log \psi_\alpha(x_\alpha) \left(\prod_{j \in \alpha \setminus i} b_j^{0,\mathtt{MF}}(x_j) \right) \left(\sum_{j \in \alpha \setminus i} R_{jk}(x_j, y_k) \right) \tag{30}$$

This update is followed by the shift (29) in order to satisfy the constraint (28), and the process is initialized with $R_{ik}(x_i, y_k) = 0$. After convergence we compute the approximate covariance according to (27).

Theorem 4 *The propagation algorithm for computing the linear response estimates of pairwise probabilities in the mean field approximation is guaranteed to converge to a unique fixed point using any scheduling of the updates.*

For a full proof we refer to the proof of theorem 6 which is very similar. However it is easy to see that for sequential updates convergence is guaranteed because (30) is the first order term of the MF equation (7) which converges for arbitrary θ.

[4]The shift can be derived by introducing a θ-dependent normalizing constant in (26), expanding it to first order in θ and using the first order terms to satisfy constraint (28).

5.2 The Bethe Approximation

In the Bethe approximation we follow a similar strategy as in the previous section for the MF approximation. First we assume that belief propagation has converged to a stable fixed point, which by [3] is guaranteed to be a local minimum of the Bethe-Gibbs free energy. Next, we expand the messages $n_{i\alpha}(x_i)$ and $m_{\alpha i}(x_i)$ up to first order in θ around the stable fixed point,

$$n_{i\alpha}(x_i) = n^0_{i\alpha}(x_i)\bigg(1 + \sum_{k,y_k} N_{i\alpha,k}(x_i, y_k)\theta_k(y_k)\bigg) \tag{31}$$

$$m_{\alpha i}(x_i) = m^0_{\alpha i}(x_i)\bigg(1 + \sum_{k,y_k} M_{\alpha i,k}(x_i, y_k)\theta_k(y_k)\bigg) \tag{32}$$

Inserting these expansions and the expansion (15) into the belief propagation equations (11,12) and matching first order terms we arrive at the following update equations for the "super-messages" $M_{\alpha i,k}(x_i, y_k)$ and $N_{i\alpha,k}(x_i, y_k)$,

$$N_{i\alpha,k}(x_i, y_k) \leftarrow \delta_{ik}\delta_{x_i y_k} + \sum_{\beta \in \mathcal{N}_i \backslash \alpha} M_{\beta i,k}(x_i, y_k) \tag{33}$$

$$M_{\alpha i,k}(x_i, y_k) \leftarrow \sum_{x_{\alpha\backslash i}} \frac{\psi_\alpha(x_\alpha)}{m^0_{\alpha i}(x_i)} \prod_{l \in \mathcal{N}_\alpha \backslash i} n^0_{l\alpha}(x_l) \sum_{j \in \mathcal{N}_\alpha \backslash i} N_{j\alpha,k}(x_j, y_k) \tag{34}$$

The super-messages are initialized at $M_{\alpha i,k} = N_{i\alpha,k} = 0$ and "normalized" as follows[5],

$$N_{i\alpha,k}(x_i, y_k) \leftarrow N_{i\alpha,k}(x_i, y_k) - \frac{1}{D_i}\sum_{x_i} N_{i\alpha,k}(x_i, y_k) \tag{35}$$

$$M_{\alpha i,k}(x_i, y_k) \leftarrow M_{\alpha i,k}(x_i, y_k) - \frac{1}{D_i}\sum_{x_i} M_{\alpha i,k}(x_i, y_k) \tag{36}$$

with D_i the number states of x_i. After the above fixed point equations have converged, we compute the response matrix $R_{ij}(x_i, x_j)$ by inserting the expansions (26,15,32) into (14) and matching first order terms,

$$R_{ij}(x_i, x_j) = \delta_{ij}\delta_{x_i x_j} + \sum_{\alpha \in \mathcal{N}_i} M_{\alpha i,j}(x_i, x_j) \tag{37}$$

We then normalize the response matrix as in (29) and compute the approximate covariances as in (27).

[5]The derivation is along similar lines as explained in the previous section for the MF case. Note also that unlike the MF case normalization is only desirable for reasons of numerical stability.

We now prove a number of useful results concerning the iterative algorithm proposed above.

Theorem 5 *If the factor graph has no loops then the linear response estimates defined in* (27) *are exact. Moreover, there exists a scheduling of the super-messages such that the algorithm converges after just one iteration (i.e. every message is updated just once).*

Proof: Both results follow from the fact that belief propagation on tree structured factor graphs computes the exact single node marginals for arbitrary θ. Since the super-messages are the first order terms of the BP updates with arbitrary θ, we can invoke the exact linear response theorem given by (17) and (18) to claim that the algorithm converges to the exact joint pairwise marginal distributions. Moreover, the number of iterations that BP needs to converge is independent of θ, and there exists a scheduling which updates each message exactly once (inward-outward scheduling). Since the super-messages are the first order terms of the BP updates, they inherit these properties. $\square$

For graphs with cycles, BP is not guaranteed to converge. We can however still prove the following strong result.

Theorem 6 *If the messages* $\{m^0_{\alpha i}(x_i), n^0_{i\alpha}(x_i)\}$ *have converged to a stable fixed point, then the update equations for the super-messages (33,34,36) will also converge to a unique stable fixed point, using any scheduling of the super-messages.*

Sketch of Proof: As a first step, we combine the BP message updates (11,12) into one set of fixed point equations by inserting (11) into (12). Next, we linearize the fixed point equations for the BP messages around the stable fixed point. We introduce a small perturbation in the logarithm of the messages: $\delta \log m_{\alpha i}(x_i) = \tilde{M}_{\alpha i}(x_i) \doteq \tilde{M}_a$ where we have collected the message index αi and the state index x_i into one "flattened" index a. The linearized equation takes the general form,

$$\log m_a + \tilde{M}_a \leftarrow \log m_a + \sum_b L_{ab}\, \tilde{M}_b \tag{38}$$

where the matrix L is given by the first order term of the Taylor expansion of the fixed point equation. Since we know that the fixed point is stable, we infer that the absolute values of the eigenvalues of L are all smaller than 1, so that $\tilde{M}_a \to 0$ as we iterate the fixed point equations.

Similarly for the super messages, we insert (33) into (34) and include the normalization (36) explicitly so that (33,34,36) collapse into one *linear* equation. We

now observe that the collapsed update equations for the super-messages are linear and of the form,

$$M_{a\mu} \leftarrow A_{a\mu} + \sum_b L_{ab} M_{b\mu} \tag{39}$$

where we introduced new flattened indices $\mu = (k, x_k)$ and where L is identical to the L in (38). The constant term $A_{a\mu}$ comes from the fact that we also expanded the node potential ψ_μ as in (15). Finally, we recall that for the linear dynamics (39) there can only be one fixed point at

$$M_{a\mu} = \sum_b \left[(I - L)^{-1}\right]_{ab} A_{b\mu} \tag{40}$$

and which exists only if $\det(I - L) \neq 0$. Finally since the eigenvalues of L are less than 1, we conclude that $\det(I - L) \neq 0$ so the fixed point exists, that the fixed point is stable, and that the (parallel) fixed point equations (39) will converge to the fixed point.

The above proves the result for parallel updates of the super-messages. However, for linear systems the Stein-Rosenberg theorem now guarantees that any scheduling will converge to the same fixed point, and moreover, that sequential updates will do so faster. □

6 Non-Iterative Algorithms for Linear Response

In section 5 we described propagation algorithms to directly compute the approximate covariances $\frac{\partial b_i(x_i)}{\partial \theta_k(x_k)}$. In this section we describe an alternative method that first computes $\frac{\partial \theta_i(x_i)}{\partial b_k(x_k)}$ and then inverts the matrix formed by $\frac{\partial \theta_i(x_i)}{\partial b_k(x_k)}$ where we have flattened $\{i, x_i\}$ into a row index and $\{k, x_k\}$ into a column index. This method is a direct extension of [4]. The intuition is that while perturbations in a single $\theta_i(x_i)$ affect the whole system, perturbations in a single $b_i(x_i)$ (while keeping the others fixed) affect each subsystem $\alpha \in A$ *independently* (see also [16]). This makes it easier to compute $\frac{\partial \theta_i(x_i)}{\partial b_k(x_k)}$ then to compute $\frac{\partial b_i(x_i)}{\partial \theta_k(x_k)}$.

First we propose minimal representations for b_i and θ_k. Notice that the current representations of b_i and θ_k are redundant: we always have $\sum_{x_i} b_i(x_i) = 1$ for all i, while for each k adding a constant to all $\theta_k(x_k)$ does not change the beliefs. This means that the matrix is actually not invertible: it has eigenvalues of 0. To deal with this non-invertibility, we propose a minimal representation for b_i and θ_i. In particular, we assume that for each i there is a distinguished value $x_i = 0$ and set $\theta_i(0) = 0$ while functionally define $b_i(0) = 1 - \sum_{x_i \neq 0} b_i(x_i)$. Now the matrix formed by $\frac{\partial \theta_i(x_i)}{\partial b_k(x_k)}$ for each i, k *and* $x_i, x_k \neq 0$ is invertible; its inverse gives us

the desired covariances for $x_i, x_k \neq 0$. Values for $x_i = 0$ or $x_k = 0$ can then be computed using (23).

6.1 The Mean Field Approximation

Taking the log of the MF fixed point equation (7) and differentiating with respect to $b_k(x_k)$, we get after some manipulation for each i, k and $x_i, x_k \neq 0$,

$$\begin{aligned}\frac{\partial \theta_i(x_i)}{\partial b_k(x_k)} =& \delta_{ik}\left(\frac{\delta_{x_i x_k}}{b_i(x_i)} + \frac{1}{b_i(0)}\right) - \\ & (1-\delta_{ik}) \sum_{\alpha \in \mathcal{N}_i \cap \mathcal{N}_k} \sum_{x_{\alpha \setminus i,k}} \log \psi_\alpha(x_\alpha) \prod_{j \in \alpha \setminus i,k} b_j(x_j) \end{aligned} \tag{41}$$

Inverting this matrix thus results in the desired estimates of the covariances (see also [4] for the binary case).

6.2 The Bethe Approximation

In addition to using the minimal representations for b_i and θ_i, we will also need minimal representations for the messages. This can be achieved by defining new quantities $\lambda_{i\alpha}(x_i) = \log \frac{n_{i\alpha}(x_i)}{n_{i\alpha}(0)}$ for all i and x_i. The $\lambda_{i\alpha}$'s can be interpreted as Lagrange multipliers to enforce the consistency constraints (10) [17]. We will use these multipliers instead of the messages in this section.

Re-expressing the fixed point equations (11,12,13,14) in terms of b_i's and $\lambda_{i\alpha}$'s only, and introducing the perturbations θ_i, we get:

$$\left(\frac{b_i(x_i)}{b_i(0)}\right)^{c_i} = \frac{\psi_i(x_i)}{\psi_i(0)} e^{\theta_i(x_i)} \prod_{\alpha \in \mathcal{N}_i} e^{-\lambda_{i\alpha}(x_i)} \qquad \text{for all } i, x_i \neq 0 \tag{42}$$

$$b_i(x_i) = \frac{\sum_{x_{\alpha \setminus i}} \psi_\alpha(x_\alpha) \prod_{j \in \mathcal{N}_\alpha} e^{\lambda_{j\alpha}(x_j)}}{\sum_{x_\alpha} \psi_\alpha(x_\alpha) \prod_{j \in \mathcal{N}_\alpha} e^{\lambda_{j\alpha}(x_j)}} \qquad \text{for all } i, \alpha \in \mathcal{N}_i, x_i \neq 0 \tag{43}$$

The divisions by the values at 0 in (42) is to get rid of the proportionality constant.

The above forms a minimal set of fixed point equations that the single node beliefs b_i's and Lagrange multipliers $\lambda_{i\alpha}$'s need to satisfy at any local minimum of the Bethe free energy. Differentiating the logarithm of (42) with respect to $b_k(x_k)$, we get

$$\frac{\partial \theta_i(x_i)}{\partial b_k(x_k)} = c_i \delta_{ik}\left(\frac{\delta_{x_i x_k}}{b_i(x_i)} + \frac{1}{b_i(0)}\right) + \sum_{\alpha \in \mathcal{N}_i} \frac{\partial \lambda_{i\alpha}(x_i)}{\partial b_k(x_k)} \tag{44}$$

remembering that $b_i(0)$ is a function of $b_i(x_i)$, $x_i \neq 0$. Notice that we need values for $\frac{\partial \lambda_{i\alpha}(x_i)}{\partial b_k(x_k)}$ in order to solve for $\frac{\partial \theta_i(x_i)}{\partial b_k(x_k)}$. Since perturbations in $b_k(x_k)$ (while keeping other b_j's fixed) do not affect nodes not directly connected to k, we have $\frac{\partial \lambda_{i\alpha}(x_i)}{\partial b_k(x_k)} = 0$ for $k \notin \alpha$. When $k \in \alpha$, these can in turn be obtained by solving, for each α, a matrix inverse. Differentiating (43) by $b_k(x_k)$, we obtain

$$\delta_{ik}\delta_{x_i x_k} = \sum_{j \in \alpha} \sum_{x_j \neq 0} C^{\alpha}_{ij}(x_i, x_j) \frac{\partial \lambda_{j\alpha}(x_j)}{\partial b_k(x_k)} \tag{45}$$

$$C^{\alpha}_{ij}(x_i, x_j) = \begin{cases} b_\alpha(x_i, x_j) - b_i(x_i) b_j(x_j) & \text{if } i \neq j \\ b_i(x_i)\delta_{x_i x_j} - b_i(x_i) b_j(x_j) & \text{if } i = j \end{cases} \tag{46}$$

for each $i, k \in \mathcal{N}_\alpha$ and $x_i, x_k \neq 0$. Flattening the indices in (45) (varying i, x_i over rows and k, x_k over columns), the LHS becomes the identity matrix, while the RHS is a product of two matrices. The first is a covariance matrix C_α where the ij^{th} block is $C^{\alpha}_{ij}(x_i, x_j)$; while the second matrix consists of all the desired derivatives $\frac{\partial \lambda_{j\alpha}(x_j)}{\partial b_k(x_k)}$. Hence the derivatives are given as elements of the inverse covariance matrix C_α^{-1}. Finally, plugging the values of $\frac{\partial \lambda_{j\alpha}(x_j)}{\partial b_k(x_k)}$ into (44) now gives $\frac{\partial \theta_i(x_i)}{\partial b_k(x_k)}$ and inverting that matrix will now give us the desired approximate covariances over the whole graph. Interestingly, the method only requires access to the beliefs at the local minimum, not to the potentials or Lagrange multipliers.

7 A Propagation Algorithm for Matrix Inversion

Up to this point all considerations have been in the discrete domain. A natural question is whether linear response can also be applied in the continuous domain. In this section we will use linear response to derive a propagation algorithm to compute the exact covariance matrix of a Gaussian Markov random field. A Gaussian random field is a real-valued Markov random field with pairwise interactions. Its energy is

$$E = \frac{1}{2} \sum_{ij} W_{ij} x_i x_j + \sum_i \alpha_i x_i \tag{47}$$

where W_{ij} are the interactions and α_i are the biases. Since Gaussian distributions are completely described by their first and second order statistics, inference in this model reduces to the computation of the mean and covariance,

$$\boldsymbol{\mu} \doteq \langle \mathbf{x} \rangle = -\mathbf{W}^{-1}\boldsymbol{\alpha} \qquad\qquad \boldsymbol{\Sigma} \doteq \langle \mathbf{x}\mathbf{x}^T \rangle - \boldsymbol{\mu}\boldsymbol{\mu}^T = \mathbf{W}^{-1} \tag{48}$$

In [14] it was shown that belief propagation (when it converges) will compute the exact means μ_i, but approximate variances Σ_{ii} and covariance Σ_{ij} between

neighboring nodes. We will now show how to compute the exact covariance matrix using linear response, which through (48) translates into a perhaps unexpected algorithm to invert the matrix $\mathbf{W}$.

First, we introduce a small perturbation to the biases, $\boldsymbol{\alpha} \rightarrow \boldsymbol{\alpha} + \boldsymbol{\nu}$ and note that,

$$\Sigma_{ij} = -\frac{\partial^2 F(\boldsymbol{\nu})}{\partial \nu_i \partial \nu_j}\Big|_{\boldsymbol{\nu}=0} = -\frac{\partial \mu_i}{\partial \nu_j}\Big|_{\boldsymbol{\nu}=0} \tag{49}$$

Our strategy will thus be to compute $\boldsymbol{\mu}(\boldsymbol{\nu}) \approx \boldsymbol{\mu}^0 - \boldsymbol{\Sigma}\boldsymbol{\nu}$ up to first order in $\boldsymbol{\nu}$. This can again be achieved by expanding the propagation updates to first order in $\boldsymbol{\nu}$. It will be convenient to collapse the 2 sets of message updates (11,12) into one set of messages, by inserting (11) into (12). Because the subsets α correspond to pairs of variables in the Gaussian random field model we change notation for the messages from $\alpha \rightarrow j$ with $\alpha = \{i, j\}$ to $i \rightarrow j$. Using the following definitions for the messages and potentials,

$$m_{ij}(x_j) \propto e^{-\frac{1}{2}a_{ij}x_j^2 - b_{ij}x_j} \tag{50}$$

$$\psi_{ij}(x_i, x_j) = e^{-W_{ij}x_i x_j} \quad \psi_i(x_i) = e^{-\frac{1}{2}W_{ii}x_i^2 - \alpha_i x_i} \tag{51}$$

we derive the update equations [6],

$$a_{ij} \leftarrow \frac{-W_{ij}^2}{W_{ii} + \sum_{k \in \mathcal{N}_i \setminus j} a_{ki}} \quad b_{ij} \leftarrow \frac{a_{ij}}{W_{ij}}\left(\alpha_i + \sum_{k \in \mathcal{N}_i \setminus j} b_{ki}\right) \tag{52}$$

$$\tau_i = W_{ii} + \sum_{k \in \mathcal{N}_i} a_{ki} \qquad \mu_i = -\frac{\alpha_i + \sum_{k \in \mathcal{N}_i} b_{ki}}{\tau_i} \tag{53}$$

where the means μ_i are exact at convergence, but the precisions τ_i are approximate [14]. We note that the a_{ij} messages do not depend on $\boldsymbol{\alpha}$ so that the perturbation $\boldsymbol{\alpha} \rightarrow \boldsymbol{\alpha} + \boldsymbol{\nu}$ will have no effect on it. Perturbing the b_{ij} messages as, $b_{ij} = b_{ij}^0 + \sum_k B_{ij,k}\nu_k$ we derive the following update equations for the "super-messages" $B_{ij,l}$,

$$B_{ij,l} = \frac{a_{ij}}{W_{ij}}(\delta_{il} + \sum_{k \in \mathcal{N}_i \setminus j} B_{ki,l}) \tag{54}$$

Note that given a solution to the above equation, it is no longer necessary to run the updates for b_{ij} (52) since b_{ij} can be computed by $b_{ij} = \sum_l B_{ij,l}\alpha_l$.

Theorem 7 *If belief propagation has converged to a stable fixed point (i.e. message updates* (52) *have converged to a stable fixed point) then the message updates*

[6] Here we used the following identity: $\int dx e^{-\frac{1}{2}ax^2 - bx} = e^{b^2/2a}\sqrt{2\pi/a}$

(54) *will converge to a unique stable fixed point. Moreover, the exact covariance matrix* $\mathbf{\Sigma} = \mathbf{W}^{-1}$ *is given by the following expression,*

$$\mathbf{\Sigma}_{il} = \frac{1}{\tau_i}\left(\delta_{il} + \sum_{k \in \mathcal{N}_i} B_{ki,l}\right) \tag{55}$$

with τ_i *given by* (53)*.*

Sketch of proof: The convergence proof is similar to the proof of theorem 6 and is based on the observation that (54) is a linearization of the fixed point equation for b_{ij} (52) so has the same convergence properties. The exactness proof is similar to the proof of theorem 5 and uses the fact that BP computes the means exactly so (49) computes the exact covariance, which is what we compute with (55) $\square$.

In [14] it was further shown that for diagonally dominant weight matrices ($|W_{ii}| > \sum_{j \neq i} |W_{ij}| \ \forall i$) convergence of belief propagation (i.e. message updates (52)) is guaranteed. Combined with the above theorem this ensures that the proposed iterative algorithm to invert $\mathbf{W}$ will converge for diagonally dominant $\mathbf{W}$. Whether the class of problems that can be solved using this method can be enlarged, possibly at the expense of an approximation, is still an open question.

The complexity is of the above algorithm is $\mathcal{O}(N \times E)$ per iteration, where N is the number of nodes and E the number of edges in the graph. Consequently, it will only improve on a straight matrix inversion if the graph is sparse (i.e. the matrix to invert has many zeros).

8 Experiments

In the following experiments we will compare 5 methods for computing approximate estimates of the covariance matrix $C_{ij}(x_i, x_j) = p_{ij}(x_i, x_j) - p_i(x_i)p_j(x_j)$:

MF: Since mean field assumes independence we have $C = 0$. This will act as a baseline.

BP: Estimates computed directly from (13) by integrating out variables which are not considered (in fact, in the experiments below the factors α consist of pairs of nodes, so no integration is necessary). Note that nontrivial estimates only exist if there is a factor node that contains both nodes. The BP messages were uniformly initialized at $m_{\alpha i}(x_i) = n_{i\alpha}(x_i) = 1$, and run until convergence. No damping was used.

MF+LR: Estimates computed from the linear response correction to the mean field approximation (section 5.1). The MF beliefs were first uniformly initialized at $b_i(x_i) = 1/D$, and run until convergence, while the response matrix $R_{ij}(x_i, x_j)$ was initialized at 0.

BP+LR: Estimates computed from the linear response correction to the Bethe approximation (section 5.2). The super-messages $\{M_{\alpha i,j}(x_i, x_j), N_{i\alpha,j}(x_i, x_j)\}$ were all initialized at 0.

COND: Estimates computed using the following conditioning procedure. Clamp a certain node j to a specific state $x_j = a$. Run BP to compute conditional distributions $b^{\text{BP}}(x_i|x_j = a)$. Do this for all nodes and all states to obtain all conditional distributions $b^{\text{BP}}(x_i|x_j)$. The joint distribution is now computed as $b_{ij}^{\text{COND}}(x_i, x_j) = b^{\text{BP}}(x_i|x_j)b^{\text{BP}}(x_j)$. Finally, the covariance is computed as,

$$C_{ij}^{\text{COND}}(x_i, x_j) = b_{ij}^{\text{COND}}(x_i, x_j) - \sum_{x_j} b_{ij}^{\text{COND}}(x_i, x_j) \sum_{x_i} b_{ij}^{\text{COND}}(x_i, x_j) \quad (56)$$

Note that C is not symmetric and that the marginal $\sum_{x_j} b_{ij}^{\text{COND}}(x_i, x_j)$ is not consistent with $b^{\text{BP}}(x_i)$.

The methods were halted if the maximum change in absolute value of all beliefs (MF) or messages (BP) was smaller than 10^{-8}.

The graphical model in the first two experiments has nodes placed on a square 6×6 grid (i.e. $N = 36$) with only nearest neighbors connected (see figure 4a). Each node is associated with a random variable which can be in one of three states ($D = 3$). The factors were chosen to be all pairs of neighboring nodes in the graph.

By clustering the nodes in each row into super-nodes exact inference is still feasible by using the forward-backward algorithm. Pairwise probabilities between nodes in non-consecutive layers were computed by integrating out the intermediate super-nodes.

The error in the estimated covariances was computed as the absolute difference between the estimated and the true values, averaged over pairs of nodes and their possible states, and averaged over 15 random draws of the network as described below. An instantiation of a network was generated by randomly drawing the logarithm of the node and edge potentials from a Gaussian with zero mean and standard deviation $\sigma_{\mathbf{node}}$ and $\sigma_{\mathbf{edge}}$ respectively.

In the first experiment we generated networks randomly with a scale $\sigma_{\mathbf{edge}}$ varying over the range $[0, 2]$ and 2 settings of the scale $\sigma_{\mathbf{node}}$, namely $\{0, 2\}$. The results (see figure 5) were separately plotted for neighboring nodes, next-to-nearest neighboring nodes and the remaining nodes, in order to show the decay of dependencies with distance.

In the next experiment we generated a single network with $\sigma_{\mathbf{edge}} = 1$ and $\{\psi_i\} = 1$ on the 6×6 square grid used in the previous experiment. The edge strengths of a subset of the edges forming a spanning tree of the graph were held

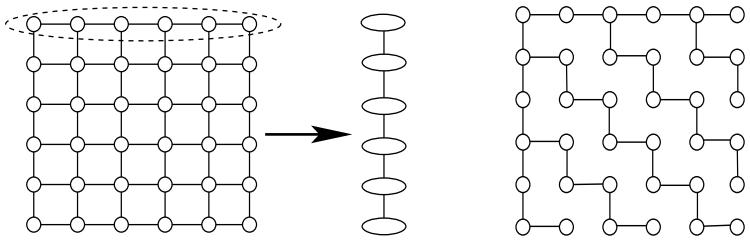

Figure 4: (a)-Square grid used in the experiments. The rows are collected into supernodes which then form a chain. (b)-Spanning tree of the nodes on the square grid used in the experiments.

fixed (see figure 4b), while the remaining edge strenghts were multiplied by a factor increasing from 0 to 2 on the x-axis. The results are shown in figure 6. Note that BP+LR and COND are exact on the tree.

Finally, we generated fully connected graphs with 10 nodes and 3 states per node. We used varying edge strengths ($\sigma_{\mathbf{edge}}$ ranging from $[0, 1]$) and two values of $\sigma_{\mathbf{node}} : \{0, 2\}$. The results are shown in figure 7. If we further increase the edge strengths in this fully connected network, we find that BP often fails to converge. We could probably improve this situation a little bit by damping the BP updates, but because of the many tight loops, BP is doomed to fail for relatively large $\sigma_{\mathbf{edge}}$.

All experiments confirm that the LR estimates of the covariances in the Bethe approximation improve significantly on the LR estimates in the MF approximation. It is well known that the MF approximation usually improves for large densely connected networks. This is probably the reason MF+LR performed better on the fully connected graph, but never as good as BP+LR or COND. The COND method performed surprisingly good, either at the same level of accuracy as BP+LR or a little bit better. It was however checked numerically that the symmetrized estimate of the covariance matrix was not positive semi-definite and that the various marginals computed from the joint distributions $b_{ij}^{\text{COND}}(x_i, x_j)$ were inconsistent with each other. In the next section we further discuss the differences between BP+LR and COND. Finally, as expected, the BP+LR and COND estimates are exact on a tree – the error is of the order of machine precision – but increases when the graph contains cycles with increasing edge strengths.

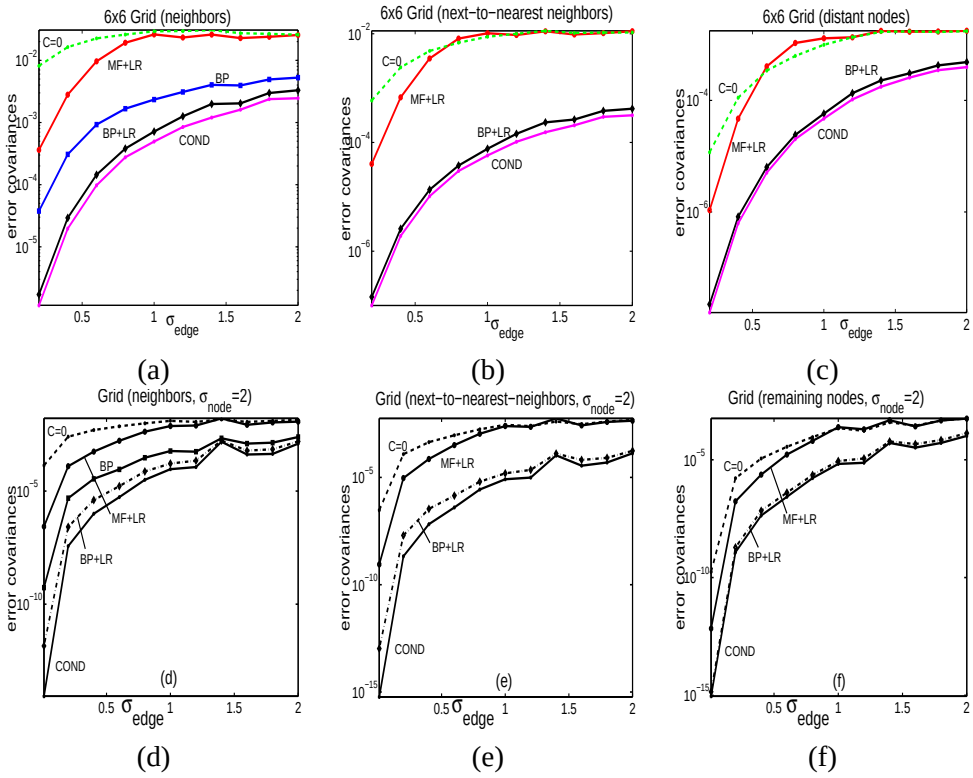

Figure 5: Absolute error for the estimated covariances for MF (C=0), MF+LR, BP, BP+LR and COND. The network is a 6×6 square grid and the log-node potentials were set to 0 in the first row (figures a,b,c) while the std. of the log-node potentials in the second row is $\sigma_{\mathbf{edge}} = 2$ (figures d,e,f). The results are averaged over all pairs of nodes, over their possible states and over 15 random instantiations of the network. The dashed line represents the baseline estimate $C = 0$ (MF) which corresponds to the statement that all nodes are independent. All figures have a logarithmic y-axis. Results are separately plotted for neighbors (a,d), next-to-nearest neighbors (b,e) and the remaining nodes (c,f). (estimates for BP are absent for (b,e) and (c,f) because BP does not provide non-trivial estimates for non-neighbors).

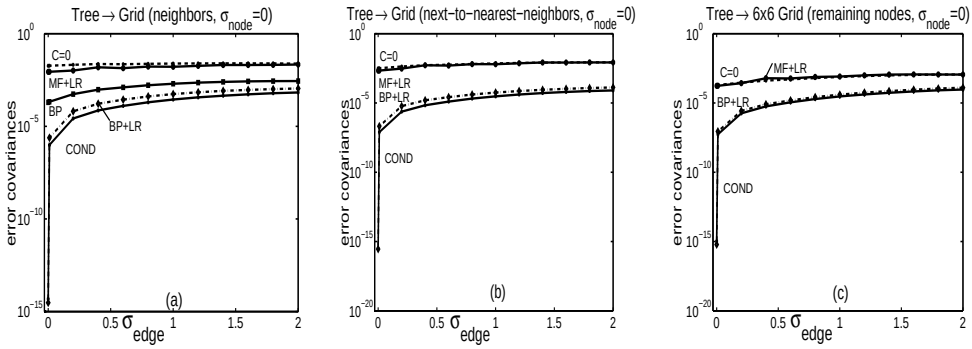

Figure 6: Plots as in figure 5. Networks were randomly drawn on a 6×6 square grid with $\sigma_{\mathbf{node}} = 0$ and $\sigma_{\mathbf{edge}} = 1$. Some edges were multiplied with a parameter ranging between $[0, 2]$ (varying over the x-axis) such that the remaining edges form a spanning tree (see figure 4b). Results were again averaged over edges, states and 15 random instantiations of the network. Results are separately plotted for neighbors (a), next-to-nearest neighbors (b) and the remaining nodes (c).

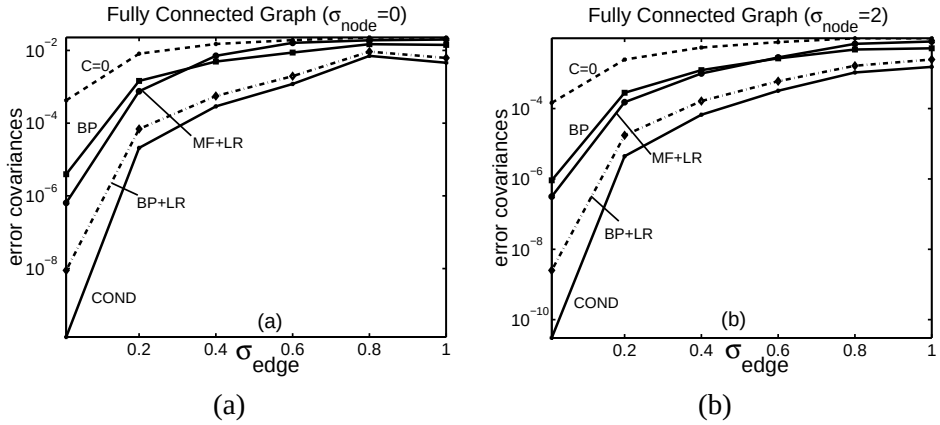

Figure 7: Absolute errors computed on a fully connected network with 10 nodes. Note that all nodes are neighbors in this case. Log-node-potentials are 0 in the figure (a) and have std. $\sigma_{\mathbf{node}} = 2$ in figure (b). Everything else as in figure 5.

9 Discussion

Loosely speaking, the "philosophy" of this paper to compute estimates of covariances is as follows (see figure 2). First we observe that the log partition function is the cumulant generating function. Next, we define its conjugate dual – the Gibbs free energy – and approximate it (e.g. the mean field or the Bethe approximation). Finally, we transform back to obtain a convex approximation to the log partition function, from which we estimate the covariances.

In this paper we have presented linear response algorithms on factor graphs. In the discrete case we have discussed the mean field and the Bethe approximations while for Gaussian random fields we have shown how the proposed linear response algorithm translates into a surprising propagation algorithm to compute a matrix inverse.

The computational complexity of the iterative linear response algorithm scales as $\mathcal{O}(N \times E \times D^3)$ per iteration, where N is the number of nodes, E the number of edges and D the number of states per node. The non-iterative algorithm scales slightly worse, $\mathcal{O}(N^3 \times D^3)$, but is based on a matrix inverse for which very efficient implementations exist. A question that remains open is whether we can improve the efficiency of the iterative algorithm when we are only interested in the joint distributions of *neighboring* nodes. On tree structured graphs we know that belief propagation computes those estimates exactly in $\mathcal{O}(E \times D^2)$, but the linear response algorithm still seems to scale as $\mathcal{O}(N \times E \times D^3)$, which indicates that some useful information remains unused. Another hint pointing in that direction comes from the fact that in the Gaussian case an efficient algorithm was proposed in [12] for the computation of variances and neighboring covariances on a loopy graph.

There are still a number of generalizations worth exploring. Firstly, instead of MF or Bethe approximations we can use the more accurate Kikuchi approximation defined over larger clusters of nodes and their intersections (see also [8]). Another candidate is the "convexified Bethe free energy" [10]. Secondly, in the case of the Bethe approximation, belief propagation is not guaranteed to converge. However, convergent alternatives have been developed in the literature [9, 19] and the non-iterative linear response algorithm can still be applied to compute joint pairwise distributions. For reasons of computational efficiency it may be desirable to develop iterative algorithms for this case. Thirdly, the presented method easily generalizes to the computation of higher order cumulants. It is straightforward (but cumbersome) to develop iterative linear response algorithms for this as well. Lastly, we are investigating whether linear response algorithms may also be applied to fixed points of the expectation propagation algorithm.

The most important distinguishing feature between the proposed LR algorithm

and the conditioning procedure described in section 8 is the fact that the covariance estimate is automatically positive semi-definite. The idea to include *global* constraints such as positive semi-definiteness in approximate inference algorithms was proposed in [11]. LR may be considered as a post-hoc projection on this constraint set (see section 4.1 and figure 3). Another difference is the lack of a convergence proof for conditioned BP runs, given that BP has converged without conditioning (convergence for BP+LR was proven in section 5.2). Even if the various runs for conditioned BP do converge, different runs might converge to different local minima of the Bethe free energy, making the obtained estimates inconsistent and less accurate (although in the regime we worked with in the experiments we did not observe this behaviour). Finally, the non-iterative algorithm is applicable to *all* local minima in the Bethe-Gibbs free energy, even those that correspond to unstable fixed points of BP. These minima can however still be identified using convergent alternatives [19, 9].

Acknowledgements

We would like to thank Martin Wainwright for discussion and the referees for valuable feedback. MW would like to thank Geoffrey Hinton for support. YWT would like to thank Mike Jordan for support.

References

[1] B.J. Frey and D.J.C. MacKay. A revolution: Belief propagation in graphs with cycles. In *Advances in Neural Information Processing Systems*, volume 10, 1997.

[2] A. Georges and J.S. Yedidia. How to expand around mean-field theory using high-temperature expansions. *J. Phys A: Math. Gen.*, 24:2173–2192, 1991.

[3] T. Heskes. Stable fixed points of loopy belief propagation are minima of the bethe free energy. In *Advances in Neural Information Processing Systems*, volume 15, Vancouver, CA, 2003.

[4] H.J. Kappen and F.B. Rodriguez. Efficient learning in Boltzmann machines using linear response theory. *Neural Computation*, 10:1137–1156, 1998.

[5] F.R. Kschischang, B. Frey, and H.A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 47(2):498–519, 2001.

[6] T. Minka. Expectation propagation for approximate Bayesian inference. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pages 362–369, 2001.

[7] K. Murphy, Y. Weiss, and M. Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 1999.

[8] K. Tanaka. Probabilistic inference by means of cluster variation method and linear response theory. *IEICE Transactions in Information and Systems*, E 6-D (7), 2003.

[9] Y.W. Teh and M. Welling. The unified propagation and scaling algorithm. In *Advances in Neural Information Processing Systems*, 2001.

[10] M.J. Wainwright, T. Jaakkola, and A.S. Willsky. A new class of upper bounds on the log partition function. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, Edmonton, CA, 2002.

[11] M.J. Wainwright and M.I. Jordan. Semidefinite relaxations for approximate inference on graphs with cycles. Technical report, CS Division, UC Berkeley, 2003. Rep. No. UCB/CSD-3-1226.

[12] M.J. Wainwright, E.B. Sudderth, and A.S. Willsky. Tree-based modeling and estimation of gaussian processes on graphs with cycles. In *Advances Neural Information Processing Systems*, volume 13, vancouver, Canada, 2000.

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

[14] Y. Weiss and W. Freeman. Correctness of belief propagation in gaussian graphical models of arbitrary topology. In *Advances in Neural Information Processing Systems*, volume 12, 1999.

[15] Y. Weiss and W. Freeman. On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs. *IEEE Transactions on Information Theory*, 47:723–735, 2001.

[16] M. Welling and Y.W. Teh. Approximate inference in boltzmann machines. *Artificial Intelligence*, 143:19–50, 2003.

[17] J.S. Yedidia, W. Freeman, and Y. Weiss. Generalized belief propagation. In *Advances in Neural Information Processing Systems*, volume 13, 2000.

[18] J.S. Yedidia, W. Freeman, and Y. Weiss. Constructing free energy approximations and generalized belief propagation algorithms. Technical report, MERL, 2002. Technical Report TR-2002-35.

[19] A.L. Yuille. CCCP algorithms to minimize the Bethe and Kikuchi free energies: Convergent alternatives to belief propagation. *Neural Computation*, 14(7):1691–1722, 2002.