Approximate Inference by Markov Chains on Union Spaces

Max Welling WELLING@ICS.UCI.EDU
Michal Rosen-Zvi MICHAL@ICS.UCI.EDU
School of Information and Computer Science, University of California Irvine, Irvine CA 92697-3425 USA

Yee Whye Teh YWTEH@EECS.BERKELEY.EDU
Computer Science Division, University of California at Berkeley, Berkeley CA94720-1776 USA.

Abstract

A standard method for approximating averages in probabilistic models is to construct a Markov chain in the product space of the random variables with the desired equilibrium distribution. Since the number of configurations in this space grows exponentially with the number of random variables we often need to represent the distribution with samples. In this paper we show that if one is interested in averages over single variables only, an alternative Markov chain defined on the much smaller "union space", which can be evolved exactly, becomes feasible. The transition kernel of this Markov chain is based on conditional distributions for pairs of variables and we present ways to approximate them using approximate inference algorithms such as mean field, factorized neighbors and belief propagation. Robustness to these approximations and error bounds on the estimates follow from stability analysis for Markov chains. We also present ideas on a new class of algorithms that iterate between increasingly accurate estimates for conditional and marginal distributions. Experiments validate the proposed methods.

1. Introduction

Graphical models have proven a powerful paradigm for modelling stochastic processes in artificial intelligence, machine learning and other related fields. Often, models contain unobserved random variables and in these cases inference becomes a core concern. For instance, learning in these "hidden variable models" is typically performed in the context of the expectation-maximization algorithm which needs inference of posterior averages in the E-step. The computational complexity of inference in graphical models, as measured by the tree-width of the graph, grows exponentially with the size of maximal cliques. This implies that exact inference is tractable only for small or highly structured graphical models. For other graphical models in which exact inference is infeasible, there are procedures that allow us to approximate the posterior distribution. Popular approximations include both optimization-based schemes like variational methods and loopy belief propagation, as well as stochastic ones like Markov chain Monte Carlo sampling.

In this work, we explore a new framework for approximate inference that applies if we only desire the marginal distributions over single variables. This new framework combines the strengths of both classes of approximate inference schemes to obtain more accurate approximations. In particular, an optimization-based scheme is first used to obtain a family of conditional probabilities that the posterior distribution should satisfy (section 4). These are then combined by running a Markov chain on the "union space" (section 2) to give more approximate approximations to the marginals. The conditionals are approximate and possibly inconsistent—we will discuss robustness and error bounds by performing a stability analysis on the Markov chains (section 3).

Let $X = \{X_1, X_2, ..., X_N\}$ be the unobserved variables, $Y = \{Y_1, \ldots, Y_M\}$ be the observed ones, and $P(X, Y)$ be a distribution over X, Y represented as a graphical model. Given an observed value y for Y, we are interested in the posterior distribution

$$P(X = x|Y = y) = \frac{P(X = x, Y = y)}{P(Y = y)} \qquad (1)$$

Computing the full list of values of $P(X = x|Y = y)$

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

for every value of x is infeasible in general. Fortunately, we often do not need access to the individual $P(X = x|Y = y)$ entries, but rather we are interested in the posterior probability that a small group of related variables X_α takes on a particular value x_α: $P(X_\alpha = x_\alpha|Y = y)$. In this paper we will be interested in marginals over single variables and pairs of neighboring variables, i.e. for $\alpha = \{i\}$ or $\{i, j\}$. To simplify notation, from here onwards we will drop the conditioning on Y and references to the variables X, writing $p(x) \doteq P(X = x|Y = y)$, $p_i(x_i) \doteq P(X_i = x_i|Y = y)$, and $P_{ij}(x_i|x_j) \doteq P(X_i = x_i|X_j = x_j, Y = y)$.

2. Markov Chains on Union Spaces

Markov chain Monte Carlo sampling is a standard technique for approximate inference in graphical models. Typically the Markov chain is defined on the *product space* of the variables,

$$\mathcal{X} = \bigotimes_{i=1}^{N} \mathcal{X}_i \tag{2}$$

where $\mathcal{X}_i$ is the space of values that the variable X_i can take on. The size of this state space scales exponentially in the number of variables N—if each variable X_i can take on D values, there are D^N states. As a result direct evaluation of the Markov chain is intractable, and it is used instead to define a procedure to obtain samples from the posterior distribution. These samples can then be used to estimate the desired functionals or marginals of the posterior distribution. Unfortunately such sampling procedures are often time consuming, and the estimates suffer from high variance which decreases only as $1/S$ with S the total number of independent samples.

Since we are only interested in marginal distributions it is possible to consider an alternate Markov chain that allows exact evaluation and directly gives the desired marginal distributions. First let us assume that we have access to the exact conditional distributions $P_{ij}(x_i|x_j)$. As these are just as expensive to obtain as the marginals themselves, we will need to approximate them, which is the topic of later sections.

The Markov chain is defined over the *union space* of the variables, given by

$$\mathcal{S} = \bigcup_{i=1}^{N} \{(i, x_i) \,:\, x_i \in \mathcal{X}_i\} \tag{3}$$

Each state (i, x_i) in the union space can be understood as choosing a particular random variable X_i and an assignment x_i. Containing only $N \times D$ elements, the union space is much smaller than the product space and as such allows for exact evaluation of the Markov chain without sampling.

Let q_i be a distribution over $\{1, 2, \ldots, N\}$ and $\pi_{i|j}$ be an ergodic transition kernel that leaves q_i invariant (see appendix A for details). Consider the following transition kernel for the union space:

$$T(i, x_i|j, x_j) = T(x_i|i, j, x_j)\; T(i|j, x_j) = P_{ij}(x_i|x_j)\pi_{i|j} \tag{4}$$

That is, given (j, x_j), we first choose the next variable i according to $\pi_{i|j}$, then we choose a value for x_i according to the conditional $P_{ij}(x_i|x_j)$. Note that we will not have $\pi_{i|j}$ depend on the actual state x_j which will prove convenient later on. We can now show that the following is an invariant distribution of T:

$$Q(i, x_i) = q_i p_i(x_i) \tag{5}$$

where $p_i(x_i)$ is the desired marginal distribution of variable i. Moreover, if T is ergodic then $q_i p_i(x_i)$ will be the unique equilibrium distribution of the Markov chain. In particular, starting from any initial distribution Q^0, we may simulate the Markov chain by direct calculation,

$$Q^{t+1}(i, x_i) \leftarrow \sum_{j,x_j} T(i, x_i|j, x_j) Q^t(j, x_j) \tag{6}$$

and we will have $Q^t \to Q$ as $t \to \infty$. Finally, we may obtain the marginal distributions using:

$$q_i^t = \sum_{x_i'} Q^t(i, x_i') \tag{7}$$

$$p_i^t(x_i) = Q^t(i, x_i)/q_i^t \tag{8}$$

for a sufficiently large value of t. This is the basic Markov chain on union space (MCUS) algorithm.

It is sometimes convenient to re-express the computations (6) directly in terms of updates to the marginal distributions. First notice that the evolution of the q_i^t distributions depends only on $\pi_{i|j}$ and crucially, not on $P_{ij}(x_i|x_j)$ (this can be seen by combining (6) and (7) and using $\sum_{x_i} P_{ij}(x_i|x_j) = 1$). Thus, we may first run a separate Markov chain to compute the equilibrium distribution q_i of $\pi_{i|j}$, and then use q_i in place of q_i^t to run the Markov chain on the marginals. In particular, we start from the initial distribution

$$Q^0(i, x_i) = q_i p_i^0(x_i) \tag{9}$$

Substituting (4) and (9) into (6), noting that $q_i^t = q_i$ for all t (since this is invariant to $\pi_{i|j}$), and dividing both sides by q_i, we get

$$p_i^{t+1}(x_i) \leftarrow \sum_j w_{j|i} \sum_{x_j} P_{ij}(x_i|x_j) p_j^t(x_j) \tag{10}$$

where the weights $w_{j|i}$ are defined as

$$w_{j|i} = \frac{\pi_{i|j} q_j}{q_i} \tag{11}$$

This definition is consistent with what is called the "time reversed" or "dual" transition matrix in the literature. In the present case it has a simple and elegant interpretation. Each node j gives a prediction of $p_i(x_i)$ based on its own marginal $p_j(x_j)$ and the conditional $P_{ij}(x_i|x_j)$. These estimates are then averaged to give the new estimate for $p_i(x_i)$ using weights $w_{j|i}$.

Note that both formulations of MCUS are equivalent, since we may recover $\pi_{j|i}$ from $w_{j|i}$ and vice versa by using (11) and noting that $w_{j|i}$ and $\pi_{i|j}$ have the same equilibrium distribution q_i. We may thus simply use the formulation that expresses our prior beliefs most easily. The choice of values for $\pi_{i|j}$ or $w_{j|i}$ will affect only the mixing time since the obtained marginals are exact. However, as we shall see next, in the approximate case this choice will also affect the approximation accuracy.

Since we do not in general have access to the exact conditional distributions $P_{ij}(x_i|x_j)$, we may instead replace them with approximate conditionals $\widetilde{P}_{ij}(x_i|x_j)$. In section 4 we describe the various approximations we can make to obtain these conditional distributions. An important result of our formulation is that our Markov chain is still well defined and will converge to some marginal distributions (assuming ergodicity), since we never required $P_{ij}(x_i|x_j)$ to be exact or even internally consistent. However, the marginals computed by (6) or (10) using approximate conditionals will not be exact and the approximation accuracy will depend on both the error in the conditionals, and our choice of $\pi_{i|j}$ or $w_{j|i}$. In section 3 we give bounds on the accuracy of the marginals in terms of the accuracy of the conditionals.

Taking the view that the Markov chain updates of (10) just take a weighted average of predictions coming from each node j, it thus makes sense to put more weight on nodes j for which reliable conditionals $\widetilde{P}_{ij}(x_i|x_j)$ are available. In particular, nodes that are close in the graph are likely to have better estimates. Also, if one has highly accurate estimates for the conditionals transiting out of a particular node J, then it makes sense to choose the $w_{J|i}$ relatively large. In section 4 we test these intuitions empirically.

3. Stability and Error Bounds

A natural question to ask at this point is whether small perturbations in the conditional probabilities can cause "large" changes in the marginal distributions, i.e. we want to study the stability of the equilibrium distribution of the Markov chain. In the following we will review some literature on the stability analysis of irreducible Markov chains described in Ipsen and Meyer (1994) and adapt them to the problem under study.

Define new indices a, b etc. by flattening the indices of the Markov chain: $a \doteq (i, x_i)$. Also define a (not necessarily small) perturbation by,

$$\tilde{T} = T + E \tag{12}$$

where T is the $DN \times DN$ dimensional transition matrix $T_{ab} \doteq P_{ij}(x_i|x_j)\pi_{i|j}$ and where the perturbation E is defined such that both T and $\tilde{T}$ are valid transition matrices. Since the fixed point of the perturbed Markov chain is again a set of marginal probabilities it is easy to see that the maximal change in the equilibrium distribution is bounded by 1, $\| Q - \tilde{Q} \| \leq 1$ where Q (with $Q_a \doteq q_i p_i(x_i)$) and $\tilde{Q}$ are the equilibrium distributions of T and $\tilde{T}$ respectively. The infinity norm $\| \cdot \|$ which we will use throughout this paper is defined as

$$\| x \| = \max_a (|x_a|) \tag{13}$$

In the following we will also use the infinity norm for matrices defined as,

$$\| M \| = \max_a \left(\sum_b |M_{ab}| \right) \tag{14}$$

Following Ipsen and Meyer (1994) we will call a Markov chain absolutely stable if there exists a finite constant (condition number) κ such that,

$$\| Q - \tilde{Q} \| \leq \kappa \| E \| \tag{15}$$

Various expressions for the condition number κ have been derived in the literature. The one we have found to give the tightest bound was $\kappa = \max_{a,b} |(I - T + Q\mathbf{1}^T)^{-1} - Q\mathbf{1}^T|_{a,b}$. To convert this into a bound for the marginals $\{p_i(x_i)\}$ we first note that,

$$\begin{aligned} q_i p_i(x_i) &\geq \left(\min_i q_i \right) p_i(x_i) \quad \Rightarrow \\ \| Q - \tilde{Q} \| &\geq \left(\min_i q_i \right) \| p - \tilde{p} \| \end{aligned} \tag{16}$$

Using this, we arrive at,

$$\| p - \tilde{p} \| \leq \kappa' \| E \| \qquad \kappa' \doteq \kappa / \left(\min_i q_i \right) \tag{17}$$

This bound on the marginal distributions tells us that small changes in the conditional distributions can only cause small errors in the marginal distributions computed by MCUS if κ' is small, i.e. the algorithm is *stable* in that case. This result is valid for any estimate of the conditional distributions with T some transition matrix based on possibly inconsistent conditional distributions and $\tilde{T}$ is a small perturbation.

There is however also a second interpretation of the bound where we set $\tilde{T}$ to be transition matrix based on the exact conditional distributions and T our estimate of it[1]. Assume furthermore that we are given a bound on the error of the transition matrix[2]: $\| T - \tilde{T} \| = \| E \| \leq \mathcal{B}$. Then, the bound in (17) will convert $\mathcal{B}$ into an error bound on the marginal distributions computed by MCUS: $\| p - \tilde{p} \| \leq \kappa' \mathcal{B}$. Thus, we can expect the error in the estimated marginal distributions to smoothly increase when the error in the estimated conditional distributions is increased.

4. Approximating the Conditionals

The proposed MCUS procedure is only practical if we can find accurate estimates for the conditional distributions $P_{ij}(x_i|x_j)$. In this section we will propose and test some methods for that purpose.

The general procedure relies on the existence of some approximate inference algorithm $\mathcal{A}$ that computes approximate marginals $\tilde{p}_i(x_i)$. If we condition node j to state $\bar{x}_j$ and run algorithm $\mathcal{A}$ on this slightly altered model, we get estimates for the conditional distributions $P_{ij}(x_i|\bar{x}_j)$. Repeating this procedure $N \times D$ times we calculate estimates for all conditional distributions. Finally, in order to employ the MCUS procedure we need to decide on the weights $w_j^{(i)}$.

In the following we will address the following three issues: 1) What should our choice of the weights $w_{j|i}$ be? 2) How does the performance of MCUS depend on the approximating algorithm $\mathcal{A}$? 3) Can we make the method of conditioning described above more efficient? All experiments are ran on binary $\{\pm 1\}$ Markov random fields with random weights.

Experiment 1: choice of weights.
To study the effect of choosing different sets of weights

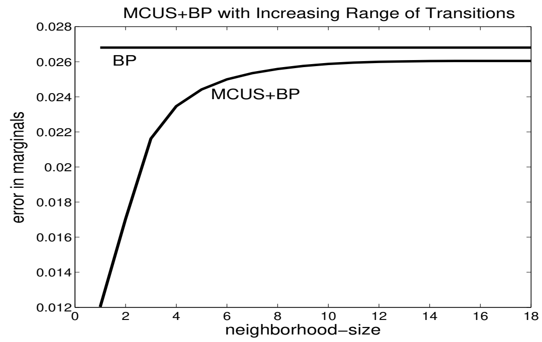

Figure 1. Dependency of MCUS performance on included transitions. On the x-axis we vary the neighborhood size of each conditional distribution (increasing to the right).

$w_{j|i}$ on the final accuracy of the approximation we performed the following experiment on a 10×10 square grid. We sampled random interactions from a Gaussian distribution with *std.* $\sigma_W = 0.5$, while biases were sampled from a Gaussian distribution with *std.* $\sigma_\alpha = 0.5$. To compute approximate conditional distributions we ran loopy BP (Yedidia et al., 2000) $N \times D$ times by separately conditioning every node on both states. The MCUS algorithm was used to compute marginal distributions $p_i(x_i)$. For each node i the weights $w_{j|i}$ were constructed by uniformly weighting nodes inside a certain neighborhood around it. Figure 1 shows the dependence of the final accuracy of MCUS as we increase this neighborhood size, starting with the Markov blanket until finally all nodes are included. We conclude that including larger neighborhoods deteriorates performance, presumably because our estimates of the conditional distributions are less reliable. This fact argues for including only the Markov blanket of a node i as non-zero entries in $w_{j|i}$ as a natural choice. In the absence of further information on the reliability of this restricted set of conditionals we set their weights equally.

Experiment 2: approximate inference methods. We study the accuracy of MCUS with conditionals computed by three different approximate inference algorithms: mean field (MF), factorized neighbors[3] (FN) and belief propagation (BP). In the experiments here we considered 2 types of graphical models: fully connected models and square grids with periodic boundary conditions[4] with a varying number of nodes. We

[1]We define T to be the approximated transition matrix because κ' is computed using properties of T, which unlike $\tilde{T}$ is the transition matrix we have access to.

[2]Assume we have an approximate inference algorithm that has an associated error bound on the marginal distributions it computes. For instance, we could use the algorithm proposed in (Leisink & Kappen, 2003) to compute such a bound. If we use this algorithm to compute conditional distributions by separately conditioning on each state of each node, then the bound on the marginals gets converted into a bound on the conditionals.

[3]In FN (Rosen-Zvi & Jordan, 2003) we iterate estimates of marginals as follows, $p_i^{t+1}(x_i) = \sum_{x_{\mathcal{N}_i}} P(x_i|x_{\mathcal{N}_i}) \prod_{j \in \mathcal{N}_i} p_j^t(x_j)$ where $\mathcal{N}_i$ is the Markov blanket of i and the conditional distributions $P(x_i|x_{\mathcal{N}_i})$ are exact.

[4]Note that this is different than the boundary condi-

sampled interactions W_{ij} and biases α_i from a uniform distribution in the interval $[-1, 1]$. The weights $w_{j|i}$ were set uniformly on the Markov blanket of each node i. Conditional distributions were computed by conditioning each node to each state and running approximate inference algorithm $\mathcal{A}$ with $\mathcal{A}$ being MF, FN or BP.

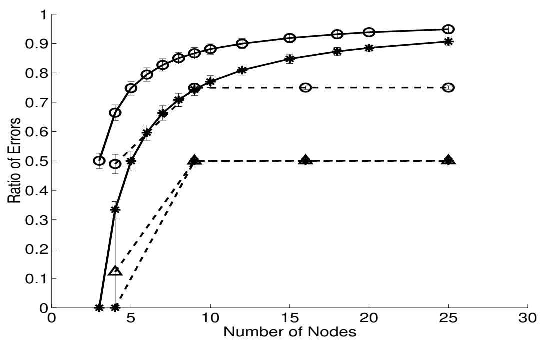

Figure 2. The ratio between the average errors of marginals computed by MCUS+$\mathcal{A}$ and the errors of marginals computed by $\mathcal{A}$. Results for both the fully connected graph (solid lines) and periodic grids (dashed lines) are shown as a function of N, the number of nodes in the graph. Circles indicate MF, triangles indicate FN and stars indicate BP. Note that the dashed lines for BP and FN overlap after $N = 9$.

In figure 2 we show the relative improvements of MCUS+$\mathcal{A}$ over $\mathcal{A}$. The results are averaged over 1000 random instantiations of the graphical models. First note that the relative improvement for MF (circles) is always smaller than that for BP and FN (stars/triangles). Further, the MCUS+BP and MCUS+FN algorithms perform significantly better than BP and FN (at least twice as good) over a wide range of network sizes. For fully connected graphs, as the number of nodes N grows the relative improvement of MCUS+$\mathcal{A}$ over $\mathcal{A}$ vanishes. The reason is that since the number of neighbors of each node increases with N, conditioning will have a diminishing effect. For square grids, the relative improvement reaches a plateau, since the number of neighbors stays constant, which implies that fixing a node to a particular state will still have an impact on it neighbors, even as $N \to \infty$. Many graphical models are of the latter kind so we expect MCUS to be a useful improvement over the corresponding approximate inference algorithms.

In Figure 3 we present the absolute errors of MCUS+$\mathcal{A}$ as a function of the errors of $\mathcal{A}$ on a 5×5 grid (stars). Note that the mean relative improvement of the MCUS

tions imposed on the square grids used in the other experiments.

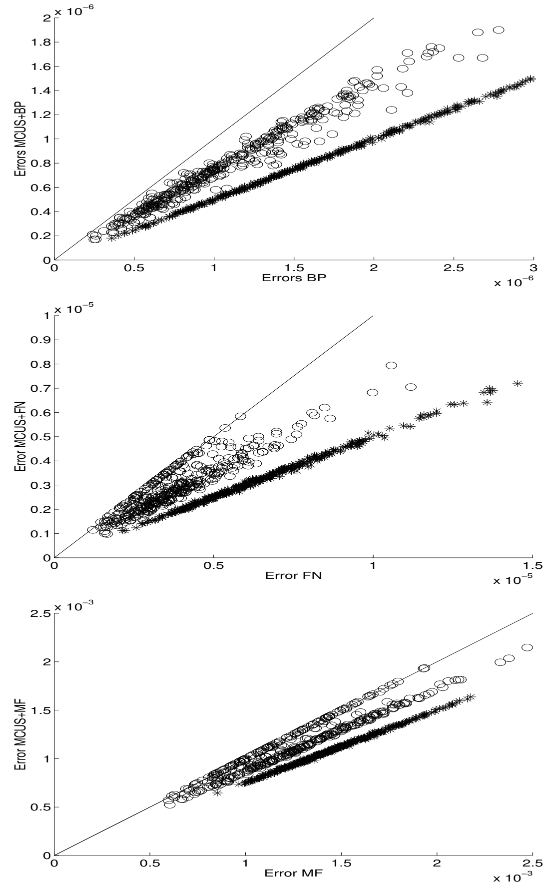

Figure 3. Errors of the MCUS+$\mathcal{A}$ marginals of single nodes as a function of the errors of $\mathcal{A}$ marginals of single nodes in a 5×5 grid (stars). Top plot shows results for $\mathcal{A} =$ BP, middle plot for FN and bottom plot for MF. The errors of the MCUS+$\mathcal{A}$ marginals of *neighboring pairs* are also presented (circles). 45-degree lines indicate equal errors.

algorithm is constant over a wide range of absolute errors, showing only small variation around the mean. This result conveys that MCUS not only performs significantly better (than BP and FN), its performance is better consistently and reliably.

The MCUS algorithm provides a straightforward way to estimate posterior marginals of pairs of nodes. One can use the estimated conditionals, $P^{\mathcal{A}}_{ij}(x_i|x_j)$, and the MCUS+$\mathcal{A}$ marginals, $p^{\texttt{MCUS+}\mathcal{A}}_i(x_i)$, and combine them to estimate the marginal of a pair as $p^{\texttt{MCUS+}\mathcal{A}}_{i,j}(x_i, x_j) = \frac{1}{2} p^{\mathcal{A}}_{i|j}(x_i|x_j) p^{\texttt{MCUS+}\mathcal{A}}_j(x_j) + \frac{1}{2} p^{\mathcal{A}}_{j|i}(x_j|x_i) p^{\texttt{MCUS+}\mathcal{A}}_i(x_i)$. We compared the result with an estimate obtained similarly, but using the marginals obtained from algorithm $\mathcal{A}$ directly (i.e. without using MCUS). Results on the 5×5 grid for neighboring pairs of nodes are shown as circles in figure 3.

The MCUS framework gave the best performance if conditionals were estimated using BP: the averaged error of BP marginals is $1.8 \cdot 10^{-6}$ while the averaged error of MCUS+BP is only $0.9 \cdot 10^{-6}$. To get an idea of the tightness of the bound proposed in section 3 we also calculated its average value for this case: $2.4 \cdot 10^{-5}$. This is unfortunately an order of magnitude above the actual error.

Experiment 3: improving the efficiency. For some applications, running algorithm $\mathcal{A}$ $N \times D$ times may be too expensive. A cheaper variation on the above method is to run algorithm $\mathcal{A}$ first to get all the (approximate) marginals, not conditioning any node to any state. Next, fixate all the quantities of interest (e.g. marginals in MF, messages in BP) outside a certain neighborhood (e.g. Markov blanket) of a node i to the values computed in this "un-clamped" run. Within this neighborhood, condition the center node i to all its possible values and run $\mathcal{A}$ on this subgraph, including but not changing the quantities outside the neighborhood. This approximation ignores the effect that conditioning has on the nodes outside the neighborhood, but is much more efficient since only local quantities need to be computed. In fact, the complexity of this algorithm scales no worse than that of the original algorithm $\mathcal{A}$. In this experiment we validate this idea by showing that the accuracy of the MCUS procedure based on these approximated conditionals saturates quickly with growing neighborhood size.

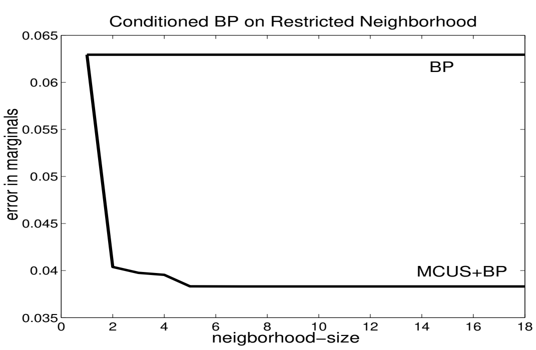

Figure 4. Dependency of MCUS performance on applying conditioning only to a restricted neighborhood.

We generated potentials on a 10×10 square grid using the same procedure as described in the previous experiment with $\sigma_W = 1$ and $\sigma_\alpha = 1$. First BP was run without conditioning to compute approximate marginals $p_i^{\mathtt{BP}}(x_i)$. Next, for each node we condition on each of its states and update the messages of BP on a restricted neighborhood only (e.g. nearest neighbors, neighbors plus next-to-nearest neighbors etc.) while keeping all messages outside that neighborhood fixed. The resulting error of the marginals resulting from MCUS based on these conditionals is shown in figure 4 as a function of increasing neighborhood size (weights $w_{j|i}$ were chosen uniform on the Markov blanket). Clearly, only updating the messages in a small neighborhood around a node is sufficient to account for most of the improved accuracy that MCUS achieves.

5. Iterating Conditionals and Marginals

In the previous section we have described a way to compute approximate conditional distributions based on some approximate inference algorithm and the method of conditioning. In this section we will describe an entirely different method to employ the MCUS algorithm to *iteratively* improve estimates of marginal distributions on single variables. We will refer to this family of algorithms as iterated MCUS or IMCUS for short.

We first re-emphasize the fact that MCUS converts estimates of conditional distributions into estimates of marginal distributions. What is needed for an iterative algorithm is a way to convert these marginal distributions back into improved estimates for the conditionals. To that end we define a neighborhood around each node i, $\mathcal{G}_i$, in which we will compute conditional distributions $P_{ij}(x_i|x_j)$. The influence of the "current state" of the nodes outside $\mathcal{G}_i$, will be approximated through some "boundary conditions" that depend on marginals $p_k(x_k)$ with $k \in \mathcal{G} \backslash \mathcal{G}_i$.

We will now discuss two different methods for that purpose, each corresponding to a variational approach to approximate inference: MF and BP. We will describe the methods based on a Markov random field with single node potentials $\psi_i(x_i)$ and interaction potentials $\psi_{ij}(x_i, x_j)$. It is straightforward to generalize this to other models.

5.1. Mean Field

In the plain vanilla MF approximation we iteratively change the node potentials by

$$\tilde{\psi}_i^{t+1}(x_i) = \psi_i(x_i) e^{\sum_{j \in \mathcal{N}_i} \sum_{x_j} \log \psi_{ij}(x_i, x_j) p_i^{\mathtt{MF},t}(x_i)} \quad (18)$$

and update the marginals according to $p_i^{\mathtt{MF},t+1}(x_i) \propto \tilde{\psi}_i^{t+1}(x_i)$. We now generalize this idea and define a neighborhood $\mathcal{G}_i$ around each node i. We assume that we have some current estimates of the marginals $p_j^{\mathtt{MF},t}(x_j)$ available on all nodes of the graph. Then, for all nodes in the neighborhood, $\mathcal{G}_i$, we change the node potentials according to (18), but only including neighbors *not* already in $\mathcal{G}_i$. Now compute a distribution

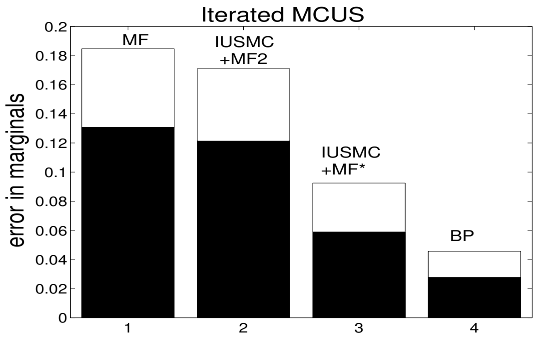

Figure 5. Comparison of two IMCUS methods with MF and BP on a fully connected graph with 10 nodes. Black bars represent mean error over 10 instantiations of the network; white top portion represents 1 standard deviation.

$P_{\mathcal{G}_i}(x_{\mathcal{G}_i})$ within the neighborhood in the usual way—a normalized product of edge and (possibly modified) node potentials. Next compute conditionals as follows,

$$P_{ij}(x_i|x_j) = \frac{\sum_{x_{\mathcal{G}_i \setminus \{i,j\}}} P_{\mathcal{G}_i}(x_{\mathcal{G}_i})}{\sum_{x_{\mathcal{G}_i \setminus j}} P_{\mathcal{G}_i}(x_{\mathcal{G}_i})}. \quad (19)$$

These conditionals are then used in MCUS to find new marginals which in turn are used to compute new conditionals etc (until convergence).

Experiment 4: IMCUS with MF.
We experimented with two versions of the algorithm, IMCUS+MF2 and IMCUS+MF*, based on different neighborhoods. The first algorithm IMCUS+MF2 uses neighborhoods $\mathcal{G}_{ij}$ consisting of i, a neighbor j of i, and the edge between them. The conditional $P_{ij}(x_i|x_j)$ is computed using the method just described with neighborhood $\mathcal{G}_{ij}$. It should be noted that in general there does not exist a solution where the estimates for the marginal distributions $\sum_{x_j} P_{\mathcal{G}_{ij}}(x_i, x_j)$ on i are consistent for all neighbors j. This fact makes MCUS a *necessary ingredient* of the algorithm. The second algorithm IMCUS+MF* uses a star-like neighborhood consisting of a node i, all of its neighbors j and all edges (ij) connecting node i to a neighbor j. Both variants always converged to reasonable estimates in all our experiments.

The results on a 10×10 square grid are shown in figure 5 (the grid was generated by an identical procedure as in section 4 with $\sigma_W = 1$ and $\sigma_\alpha = 0.5$). We clearly see a significant improvement in performance if we increase the neighborhood and there is hope that by choosing for instance spanning trees (instead of stars) the performance may approach or exceed that of BP.

5.2. Belief Propagation

In the usual formulation of BP we deal with messages. However, since there is no procedure that converts marginals into messages, we need to switch to a formulation of BP that uses marginals directly. In Teh and Welling (2002) the "Unified Propagation and Scaling" (UPS) algorithm was proposed as a convergent alternative to BP. The idea is that the graph is divided into (possibly overlapping) tree-structured subgraphs. Given a tree-structured neighborhood $\mathcal{G}_i$ around i, iterative scaling (IS) is used to compute a joint distribution on $\mathcal{G}_i$ subject to the constraints that the marginals on the nodes on the boundary of this subgraph remain fixed to their values calculated in a previous iteration $p_i^{\mathsf{UPS},t}(x_i)$. Thus, like in the case of MF, this may be viewed as an alternative procedure to incorporate the influence of the "outside nodes" in $\mathcal{G} \setminus \mathcal{G}_i$ into the neighborhood $\mathcal{G}_i$.

From the joint distribution on $\mathcal{G}_i$ we can now compute conditional distributions $P_{ij}(x_i|x_j)$ and run MCUS (using these conditionals) to get updated marginals etc. It is however not hard to show that the fixed points of UPS imply the fixed points of this IMCUS algorithm, irrespective of the way the weights $w_{j|i}$ are distributed over the nodes in $\mathcal{G}_i$ (the reverse statement seems harder to prove). This fact is not necessarily true for the generalization where the sub-graph $\mathcal{G}_i$ is arbitrary (i.e. not tree-structured). If certain nodes are members of different neighborhoods, the UPS procedure is no longer guaranteed to converge, due to the fact that the updates in the different neighborhoods can not be made consistent (this effect has indeed been verified experimentally). To resolve this we propose the following IMCUS procedure. First collect conditionals $P_{ij}(x_i|x_j)$, with j possibly residing in different neighborhoods; subsequently run the MCUS and iterate. Although this is still work in progress we expect this procedure to converge to increasingly accurate estimates of the marginals as we increase the neighborhoods $\mathcal{G}_i$.

To summarize, we have shown that a combination of MCUS and a method to incorporate the influence of outside marginals into a neighborhood $\mathcal{G}_i$ for each node i, has resulted in a very general class of approximate inference algorithms[5]. The convergence properties and accuracy of the various algorithms remains to be studied in more depth.

[5]The factorized neighbors (FN) algorithm (Rosen-Zvi & Jordan, 2003) can be viewed in this light as well. In that case however, the IMCUS procedure does not seem to generate a *new* algorithm because fixed points of FN always imply fixed points of IMCUS.

6. Discussion

Markov chains on union spaces have an interesting analogue in the literature on "reversible jump MCMC" which was developed for Bayesian model selection (Green, 1995). Initial attempts to sample from spaces with different numbers of parameters (corresponding to different models) were formulated on product spaces before the union space formulation was discovered. We want to emphasize however that in that case the transition probabilities are exact and the posterior probabilities are still estimated through sampling, whereas in this paper we approximate the transition probabilities and evolve the marginal distributions without sampling.

An interesting question that we leave for future research is whether the MCUS method can be extended to estimate marginals on larger clusters of nodes, e.g. pairs of neighboring nodes. The set of transition probabilities that needs to be approximated is now much larger and includes transitions between any marginal distribution on subsets of nodes that one wants to represent.

We anticipate that the presented methods may also have applications outside the field of approximate inference.

A. Spectral Analysis for Markov Chains

The eigenvalues of a stochastic (transition) matrix must satisfy $|\lambda_i| \leq 1$ (Bremaud, 1998). Irreducibility means that there is only one eigenvalue with $\lambda = 1$, corresponding to the equilibrium distribution. Aperiodicity means that there are no eigenvalues with $|\lambda| = 1$ other than $\lambda = 1$. Ergocity means that the MC is both irreducible and aperiodic.

The power-method iterates $p_i^{t+1} = \sum_j T_{ij} p_j^t$ until convergence. If the MC is ergodic this method converges to the unique equilibrium distribution. If the MC is reducible it converges to an arbitrary linear combination of eigenvectors with $\lambda = 1$. If the chain is periodic it will switch between the eigenvectors with eigenvalues on the complex circle. However, the latter is easily remedied by changing $T' = (1-\alpha)T + \alpha \mathbf{I}$ which pulls all eigenvalues with $|\lambda| = 1, \quad \lambda \neq 1$ inside the complex circle which implies that the power method for T' converges to the eigen-vector of T with $\lambda = 1$. Finally, the subdominant eigenvalue determines the speed of convergence (or mixing rate) of the power method.

The transition kernel $T(i, x_i | j, x_j) = \pi_{i|j} P_{ij}(x_i, x_j)$ has the structure of a generalized Hamadard product. All eigenvalues of π will also be eigenvalues of T. To see that, let $\{v^{(k)}\}$ be the left-eigenvectors of π with eigen-value $\lambda^{(k)}$ and note that $\mathbf{1}(x_i)$, the vector of all ones, are left eigen-vectors of $P_{ij} \;\; \forall i,j$ with eigen-values 1. Then, $\sum_{i,x_i} v_i^{(k)} \mathbf{1}(x_i) \pi_{i|j} P_{ij}(x_i|x_j) = \lambda^{(k)} v_j^{(k)} \mathbf{1}(x_j)$. This has the important implication that spectral properties of π, such as irreducibility and aperiodicity carry over to T.

References

Bremaud, P. (1998). *Markov chains, Gibbs fields, Monte Carlo simulation, and queues*. Springer.

Green, P. (1995). Reversible jump MCMC computation and Bayesian model determination. *Biometrika*, *82*, 711–732.

Ipsen, I., & Meyer, C. (1994). Uniform stability of Markov chains. *SIAM Journal on Matrix Analysis and Applications*, *15*, 1061–1074.

Leisink, M., & Kappen, B. (2003). Bound propagation. *J. of Artificial Intelligence Research*, *19*, 139–154.

Rosen-Zvi, M., & Jordan, M. (2003). *Approximate inference and the DLR equations* (Technical Report). Technical Report, Computer Science Division, University of California, Berkeley.

Teh, Y., & Welling, M. (2002). The unified propagation and scaling algorithm. *Advances in Neural Information Processing Systems*.

Yedidia, J., Freeman, W., & Weiss, Y. (2000). Generalized belief propagation. *Advances in Neural Information Processing Systems*.