An Edge Deletion Semantics for Belief Propagation

Arthur Choi, Adnan Darwiche

Computer Science Department
University of California, Los Angeles
Los Angeles, CA 90095, USA

Abstract

Iterative belief propagation is an influential method for approximate inference in probabilistic graphical models, perhaps the most influential method of the last decade. Given its wide-spread applicability in various domains, there has been a great interest in developing semantics for this method to both characterize and control the quality of its approximations. We present in this paper a new semantics for belief propagation, formalizing it as a method of exact inference on a simplified model that has been obtained by deleting edges from the original. When we delete an edge, however, we lose a model dependency, which we can compensate for by adding auxiliary parameters to the model. We show that the iterations of belief propagation are searching for such auxiliary parameters in a model which results from deleting every network edge. This semantics leads to a number of questions: Can we delete fewer than every model edge? Further, which edges should we delete and which should we let remain? The answers to these questions lead to a generalization of belief propagation based on edge deletion, which we present here, allowing one to trade approximation quality with computational resources. This edge deletion perspective sheds new light on belief propagation approximations, and further enables an effective procedure for finding improved approximations through a simple process of edge recovery.

Key words: Bayesian networks, probabilistic graphical models, approximate inference, belief propagation, generalized belief propagation, partition function, free energy approximations, Bethe free energy, edge deletion

⋆ This paper is based on prior publications (Choi and Darwiche, 2006a, 2008a).

Email addresses: aychoi@cs.ucla.edu (Arthur Choi), darwiche@cs.ucla.edu (Adnan Darwiche).

1 Introduction

Classical algorithms for exact probabilistic inference on graphical models have a complexity which is parameterized by the network topology (Jensen et al., 1990; Lauritzen and Spiegelhalter, 1988; Zhang and Poole, 1996; Dechter, 1996; Darwiche, 2001). In particular, it is well known that exact inference can be performed exponential only in the treewidth of a given network, where treewidth is a graph theoretic parameter that measures network connectivity. Therefore, models with high treewidth (and no local structure, Chavira and Darwiche, 2006) can be inaccessible to these methods, necessitating the use of approximation algorithms. Iterative belief propagation (IBP), also known as loopy belief propagation (Pearl, 1988; Yedidia et al., 2003), is one such algorithm that has been extremely influential in certain classes of applications. For instance, IBP has spawned approaches capable of solving particularly difficult instances of the satisfiability problem (Braunstein et al., 2005), and has shown to be an effective approach to a variety of computer vision tasks (Szeliski et al., 2006), particularly stereo vision (Felzenszwalb and Huttenlocher, 2006; Sun et al., 2005). Its biggest impact has been in the field of information theory, where revolutionary algorithms for decoding error-correcting codes have shown to be instances of iterative belief propagation in a Bayesian network (Frey and MacKay, 1997; McEliece et al., 1998).

We propose in this paper a new perspective on this influential algorithm in Bayesian networks, viewing it as an algorithm run in a simplified approximation of the original network. In fact, we show that IBP corresponds to a tree-structured approximation, where exact inference has computational complexity that is linear in the size of the model. This correspondence holds even in the case of a fully-disconnected approximation (which is vacuously a tree). More specifically, the approximate network results from deleting enough edges from the original network, where the loss of each edge is offset by introducing free parameters into the approximate network. We show that the iterations of belief propagation can be understood as searching for values of these free parameters that satisfy intuitive conditions that we formally characterize.

This edge deletion semantics of IBP leads to a number of implications. On the theoretical side, it provides a new, intuitive characterization of the fixed points of IBP. On the practical side, it leads to a concrete framework for improving the quality of approximations returned by IBP. In particular, since IBP corresponds to deleting enough edges to yield a polytree,[1] one wonders whether recovering some of the deleted edges can improve the quality of approximations; see Figure 1. The answer is yes, as we show later. Indeed, the

[1] In the Bayesian network literature, a polytree is a DAG whose underlying undirected graph is also acyclic.

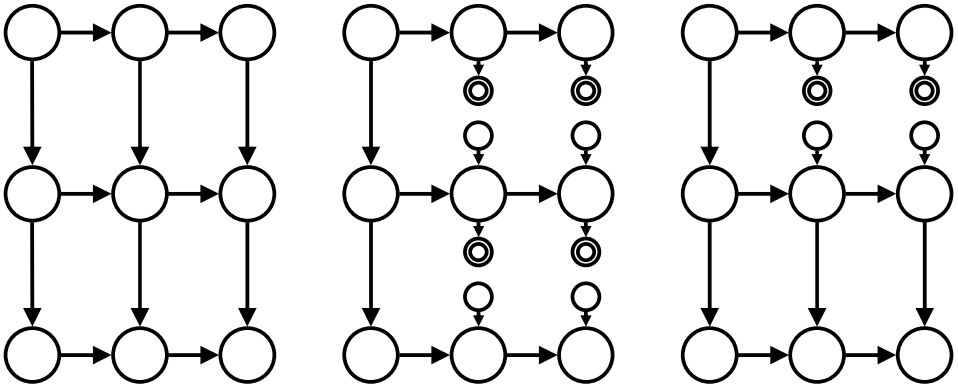


Fig. 1. A network (left), a polytree approximation (center), and a more structured approximation (right).

edge recovery proposal can be quite practical if it leads to a more structured approximation that is still feasible for exact inference. This leads to another question: What edges are the most promising to recover? For this, we appeal to the semantics of edge deletion which suggest a criterion for recovering edges based on mutual information. We discuss the properties of this recovery method and provide empirical results, showing how it can identify a *small* set of edges that can effectively improve the quality of an approximation without impacting much the complexity of inference. Our method also identifies edges whose recovery may increase inference complexity, without having a justifiable impact on the quality of approximations.

1.1 Overview

For the reader who may not already be familiar with belief propagation, or the significant body of research that is based on it, we seek in this paper to provide a thorough introduction, based on an edge deletion semantics, that should be accessible to those with a basic knowledge of Bayesian networks. For a classical introduction to belief propagation in Bayesian networks, we suggest (Pearl, 1988; Murphy et al., 1999); for a relatively more recent introduction, for undirected models, we recommend (Yedidia et al., 2003).

For the reader who is already familiar with belief propagation and some of its generalizations, we seek to present a new perspective on belief propagation that allows us to better understand belief propagation and design more accurate approximations. Along the way, we hope to illustrate that an edge-deletion semantics is expressive enough to reveal a number of familiar properties that were previously visible only from other disparate perspectives. At the same time, we hope to illustrate that this semantics is simple enough to allow us to easily identify improved approximations, a task which is often non-trivial in other generalized instances of belief propagation.

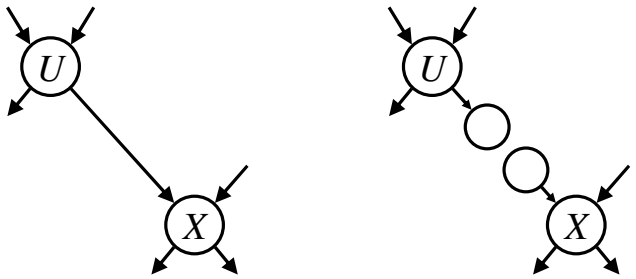


Fig. 2. We delete an edge $U \rightarrow X$ by (1) replacing parent variable U with a clone variable $\hat{U}$ and (2) introducing a soft evidence variable $\hat{S}$.

We begin in Section 2 by introducing a general framework for approximating a Bayesian network by deleting edges. In Section 3, we propose a new semantics for belief propagation based on this framework of edge deletion. We show that belief propagation falls as a degenerate case in this framework, in the extreme case where *every* model edge is deleted. We go on to propose an edge recovery procedure for identifying more accurate approximations. In Section 4, we review the Bethe free energy approximation and its relationship to iterative belief propagation, and show how it also arises as a result of deleting edges. We provide an empirical analysis in Section 5, review related work in Section 6, and finally conclude in Section 7. The Appendix includes proofs of theorems, as well as a formulation of edge deletion in undirected models.

2 Deleting a Directed Edge

Let $U \rightarrow X$ be an edge in a Bayesian network, and suppose that we wish to delete this edge to make the network more amenable to exact inference algorithms. This deletion will introduce two problems. First, variable X will lose its direct dependence on parent U. Second, variable U may lose evidential information received through its child X. To address these problems, our proposal for edge deletion introduces two auxiliary variables to help compensate for these lost dependencies.

Definition 1 (Edge Deletion) *Let $U \rightarrow X$ be an edge in a Bayesian network N. We say that the edge $U \rightarrow X$ is deleted when it results in a network that is obtained from N as follows:*

- *The edge $U \rightarrow X$ is removed from the graph.*
- *A new variable $\hat{U}$ replaces variable U as a parent of X, where $\hat{U}$ is a clone of U, having the same states as U.*
- *A binary variable $\hat{S}$ replaces X as a child of U, where $\hat{S}$ is instantiated, acting as soft evidence on U.*

Our definition of edge deletion is illustrated in Figure 2. The first auxiliary variable introduced is a variable $\hat{U}$, which compensates for the lost influence that parent U had on its child X. Thus, variable $\hat{U}$ is a clone of variable U. The second auxiliary variable is an *instantiated* variable $\hat{S}$, which compensates for the lost influence that evidence connected to variable X had on variable U.[2] Since auxiliary variable $\hat{S}$ represents evidence, and will always be fixed to some state $\hat{s}$, we assume that variable $\hat{S}$ is binary. Moreover, since auxiliary variable $\hat{U}$ is a clone, it has the same states as variable U. Thus, we leave the conditional probability tables (CPT) of variable X unchanged from the original model, except that it now depends on the cloned parent $\hat{U}$.

The deletion of an edge $U \rightarrow X$ introduces new parameters into the network, as we must now provide CPTs for the new variables $\hat{U}$ and $\hat{S}$. Variable $\hat{U}$, a root node in the network, has a CPT $\Theta_{\hat{U}}$ representing the prior marginal on variable $\hat{U}$. Variable $\hat{S}$, a leaf node in the network, has a CPT $\Theta_{\hat{S}|U}$ representing the conditional probability of $\hat{S}$ given U. Since evidence variable $\hat{S}$ is instantiated to a state $\hat{s}$, we only need to specify the CPT parameters $\theta_{\hat{s}|u}$ for each state u of U, where the complementary parameters $\theta_{\neg\hat{s}|u}$ are simply set to $1 - \theta_{\hat{s}|u}$.

For each deleted edge, we thus need to specify two vectors, each with as many elements as there are states u in variable U. We shall use the vectors $PM(\hat{U})$ and $SE(U)$ to denote the required parameters, which we shall refer to as *edge parameters*. More specifically, $PM(\hat{U}=\hat{u})$ denotes the CPT parameter $\theta_{\hat{u}}$, and $SE(U=u)$ denotes the CPT parameter $\theta_{\hat{s}|u}$. We are particularly interested in the pairs $PM(\hat{U}=u), SE(U=u)$ where variables $\hat{U}$ and U are set to the same state u.

Note again that we view the auxiliary variable $\hat{S}$ as injecting a *soft evidence* on variable U, whose strength is defined by the parameters $SE(U)$. In contrast to hard evidence, which would fix U to a particular state u, soft evidence can increase our belief in u, but not necessarily to the point of certainty. Moreover, for queries that are conditioned on evidence $\hat{s}$, only the relative ratios of parameters $SE(U)$ matter, not their absolute values. For more on soft evidence, see (Pearl, 1988; Chan and Darwiche, 2005).

Now that we have defined our notion of deleting an edge, we need to answer the following two questions to specify an approximation scheme.

- *How do we parametrize edges?* The extent to which this proposal is successful will depend on the specific edge parameters that are used. In particular, the quality of the resulting approximation depends on the extent we can

[2] Our proposal for deleting an edge is an extension of the one given by (Choi et al., 2005). It proposed the addition of a clone variable but missed the addition of an evidence variable, which is vital for the purposes of subsuming IBP.

compensate for the dependencies lost when deleting edges.

- *Which edges do we delete?* This choice impacts the quality of the resulting approximation, but its computational complexity as well. Indeed, we have a spectrum of approximations from which to choose from. On one end, where every edge is deleted, inference is trivial but provides the coarsest approximation. On the other end, when no edge is deleted, we have the complexity and accuracy of exact inference.

Moreover, we may want to choose different parameters and different edges to delete, depending on what evidence has been observed, or what query we are approximating (e.g., node marginals or MAP explanations). Indeed, we shall take this query-specific approach, seeking to approximate a model for a given query, rather than approximating a model once for all possible queries.

2.1 An Example

Before we proceed, consider as an example Figure 3 which defines a simple network where a single edge has been deleted. Before we can use the network N' to approximate queries on N, we must first specify its edge parameters. For example, we can use the parameters:

$\hat{S}$	A	$\Theta_{\hat{S}\mid A} = SE(A)$
true	true	0.3438
true	false	0.6562

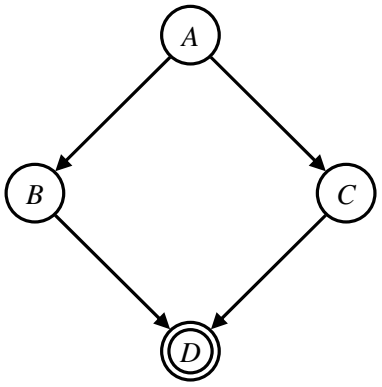
$\hat{A}$	$\Theta_{\hat{A}} = PM(\hat{A})$
true	0.8262
false	0.1738

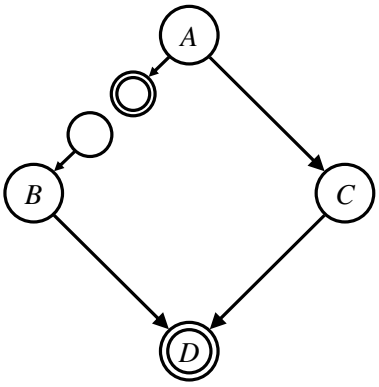
where we omit the redundant CPT parameters for variable $\hat{S}$ for the case $\hat{S}=\mathsf{false}$. In the approximate network N', we set the same evidence $\mathbf{e} : D=\mathsf{true}$ that we set in N, but we also set the auxiliary variable $\hat{S}$ in the augmented evidence $\mathbf{e}' : D=\mathsf{true}, \hat{S}=\mathsf{true}$.

This particular choice of edge parameters then gives us the following approximations to node marginals (up to four significant digits):

	$Pr'(A \mid \mathbf{e}')$	$Pr'(B \mid \mathbf{e}')$	$Pr'(C \mid \mathbf{e}')$	$Pr'(D \mid \mathbf{e}')$
true	0.7135	0.3442	0.5866	1.0
false	0.2865	0.6558	0.4134	0.0

For comparison, the exact node marginals are given by:





A	Θ_A
true	0.8
false	0.2

A	B	$\Theta_{B\|A}$
true	true	0.8
true	false	0.2
false	true	0.4
false	false	0.6

A	C	$\Theta_{C\|A}$
true	true	0.5
true	false	0.5
false	true	1.0
false	false	0.0

B	C	D	$\Theta_{D\|BC}$
true	true	true	0.1
true	true	false	0.9
true	false	true	0.3
true	false	false	0.7
false	true	true	0.9
false	true	false	0.1
false	false	true	0.8
false	false	false	0.2

Fig. 3. A Bayesian network (upper left) conditioned on evidence $\mathbf{e} : D{=}\mathsf{true}$ and an approximate network (upper right) where we delete edge $A \to B$, and condition on augmented evidence $\mathbf{e}' : D{=}\mathsf{true}, \hat{S}{=}\mathsf{true}$. Note that $\Theta_{B|\hat{A}} = \Theta_{B|A}$.

	$Pr(A \mid \mathbf{e})$	$Pr(B \mid \mathbf{e})$	$Pr(C \mid \mathbf{e})$	$Pr(D \mid \mathbf{e})$
true	0.6947	0.3579	0.5789	1.0
false	0.3053	0.6421	0.4211	0.0

3 Edge Deletion and Belief Propagation

Given a network N and evidence $\mathbf{e}$, our proposal is then to approximate this network with a simpler network N' that results from deleting some number of edges $U \to X$ as defined in Definition 1. Moreover, when performing inference on network N', we will condition on the augmented evidence $\mathbf{e}'$ composed of

the original evidence $\mathbf{e}$ and each piece of auxiliary evidence $\hat{S}=\hat{s}$ introduced when deleting edges. More formally, if Pr and Pr' are the distributions induced by networks N and N', respectively, we will use the conditional distribution $Pr'(.|\mathbf{e}')$ to approximate $Pr(.|\mathbf{e})$.

We cannot commence with the above proposal, however, without first specifying a way to choose the parameters $PM(\hat{U})$ and $SE(U)$ for each deleted edge $U \rightarrow X$. There are two issues here. First, what parameters should we seek? Second, how do we compute them? As to the first question, consider the scenario where we delete a single edge $U \rightarrow X$ that splits a network into two disconnected subnetworks. In this case, we can in fact identify edge parameters that compensate for the deleted edge precisely, allowing us to compute exact marginals over the variables in each subnetwork. Note, however, that joint marginals involving variables in both subnetworks will remain approximate, since they become independent after deletion.

We can in fact characterize a class of approximate networks, where possibly many edges are deleted, that yield exact results for those particular networks where a deleted edge splits a network into two. Exactness in particular holds for polytrees, as deleting any edge in a polytree splits the network into two. In general, these networks, simplified by edge deletion, will give rise to a generalization of the belief propagation algorithm, and in particular the approximation algorithm for networks with undirected cycles. We thus refer to these edge deletion networks as ED-BP approximations, which are characterized as follows.

Condition 1 (ED-BP) *Let N be a Bayesian network and N' be the result of deleting edges $U \rightarrow X$ from N. Edge parameters of an* ED-BP *approximation satisfy the following conditions:*

$$Pr'(U=u \mid \mathbf{e}') = Pr'(\hat{U}=u \mid \mathbf{e}'), \tag{1}$$
$$Pr'(U=u \mid \mathbf{e}' \setminus \hat{s}) = Pr'(\hat{U}=u) \tag{2}$$

for all states u, and for all edges $U \rightarrow X$ deleted.

Lemma 1 *Let N be a Bayesian network and N' be the result of deleting an edge $U \rightarrow X$ that splits N into two disconnected subnetworks. Further, say that this deletion partitions the original network variables into two disjoint sets: $\mathbf{X}_U$ containing variable U, and $\mathbf{X}_X$ containing variable X. Then the edge parameters of network N' satisfy Condition 1 if and only if the joint distributions for each subnetwork are exact:*

$$Pr'(\mathbf{X}_U \mid \mathbf{e}') = Pr(\mathbf{X}_U \mid \mathbf{e}), \qquad Pr'(\mathbf{X}_X \mid \mathbf{e}') = Pr(\mathbf{X}_X \mid \mathbf{e}).$$

Consider an approximate network where edge parameters satisfy Condition 1.

Equation 1 tells us that variables $\hat{U}$ and U have the same posterior marginals. This is not surprising given the semantics of auxiliary variable $\hat{U}$, which say that $\hat{U}$ is a clone of variable U. Equation 2 is perhaps less obvious, but consider the situation where deleting an edge $U \rightarrow X$ splits a network into two disconnected subnetworks. In this case, any variable in $\mathbf{X}_X$ that was set as evidence will be disconnected from variable U. Equation 2 tells us that, in a network satisfying Condition 1, this lost evidence is compensated for by the auxiliary evidence $\hat{s}$. That is, Equation 2, in light of Equation 1, says that observing $\hat{s}$ has the same impact on variable U as observing all evidence has on clone $\hat{U}$:

$$\frac{Pr'(U=u \mid \mathbf{e}')}{Pr'(U=u \mid \mathbf{e}' \setminus \hat{s})} = \frac{Pr'(\hat{U}=u \mid \mathbf{e}')}{Pr'(\hat{U}=u)}.$$

Assuming that an edge splits the network, only variables in $\mathbf{X}_X$ will have an impact on clone $\hat{U}$ in N'. Hence, in this case, the auxiliary evidence $\hat{s}$ does indeed summarize the effect of all variables in $\mathbf{X}_X$ that were set as evidence.

If deleting an edge does not split the network, not all evidence connected to X will be disconnected from U. Hence, Equation 2 of Condition 1 can be seen as overcompensating. It is indeed this observation which will be the basis of our heuristic for recovering edges, to be discussed later. Further, we will show that the stated conditions characterize the fixed points of IBP (and some of its generalizations) on Bayesian networks.

3.1 Searching for Edge Parameters

Now that we know what edge parameters to look for, how do we find them? The following theorem provides an answer.

Theorem 1 *Let N be a Bayesian network and N' be the result of deleting edges $U \rightarrow X$ from N. Condition 1 holds in network N' if and only if its edge parameters satisfy the following conditions:*

$$PM(\hat{U}=u) = \alpha \frac{\partial Pr'(\mathbf{e}')}{\partial SE(U=u)} \tag{3}$$

$$SE(U=u) = \alpha \frac{\partial Pr'(\mathbf{e}')}{\partial PM(\hat{U}=u)}, \tag{4}$$

where $\alpha > 0$ is a normalizing constant.

Since $PM(\hat{U})$ represents the prior distribution $\Theta_{\hat{U}}$, it should normalize to sum to one. Since $SE(U)$ represents soft evidence parameters $\theta_{\hat{s}|u}$, it does not need to sum to one, although for symmetry we shall assume that it does. Note further that these partial derivatives can be computed relatively efficiently in traditional inference frameworks (Darwiche, 2003; Park and Darwiche, 2004).

In Appendix B, we provide a probabilistic semantics for these edge parameters, as well as identify the value of the normalization constants α (which happen to be the same value in both equations).

Theorem 1 suggests the following iterative method for identifying edge parameters. First, we start with a network N'_0 with a corresponding distribution Pr'_0, where all edge parameters $PM_0(\hat{U})$ and $SE_0(U)$ are initialized, say, uniformly. For each iteration $t > 0$, the parameters for edges in N'_t will be determined by performing exact inference on the approximate network N'_{t-1}, from the previous iteration:

$$PM_t(\hat{U}=u) = \alpha \frac{\partial Pr'_{t-1}(\mathbf{e}')}{\partial SE_{t-1}(U=u)} \tag{5}$$

$$SE_t(U=u) = \alpha \frac{\partial Pr'_{t-1}(\mathbf{e}')}{\partial PM_{t-1}(\hat{U}=u)} \tag{6}$$

where we use α as a normalizing constant. We then keep iterating until all edge parameters converge to a fixed point (if ever).[3] Note that all edge parameters can be computed using a single jointree propagation and then updated in parallel; again, see (Darwiche, 2003; Park and Darwiche, 2004) for computing partial derivatives.

Algorithm 1 summarizes the above procedure, which we call ED-BP, an iterative method for identifying the ED-BP approximations satisfying Condition 1. As an example, consider again the Bayesian network in Figure 3, where we want edge parameters for a single deleted edge. Algorithm ED-BP produces a sequence of edge parameters, as follows:

	$A/\hat{A}$	$t=0$	$t=1$	$t=2$	$t=3$	$t=4$
$PM_t(\hat{A})$	true	0.5000	0.8142	0.8257	0.8262	0.8262
	false	0.5000	0.1858	0.1743	0.1738	0.1738
$SE_t(A)$	true	0.5000	0.3496	0.3440	0.3438	0.3438
	false	0.5000	0.6504	0.6560	0.6562	0.6562

(up to four significant digits). At iteration $t = 0$, we initialize our edge parameters uniformly, giving us an initial network N'_0. We compute parameters for the next network N'_1 by performing exact inference in N'_0, according to Equations 5 and 6. We repeat, computing parameters for N'_2 from N'_1, and so on, until at iteration $t = 4$, we notice that our parameters are the same

[3] As we shall see in the following section, this iterative procedure corresponds, in a degenerate case, to message passing in IBP. As such, it inherits to an extent the convergence properties of IBP (Elidan et al., 2006). However, in Section 5, we show that the edge deletion perspective can have better convergence behavior in practice.

Algorithm 1 ED-BP

input:

N: a Bayesian network

$\mathbf{e}$: an instantiation of some variables in network N

$\mathcal{E}$: a set of edges in N

output: an approximate network N' satisfying Condition 1

main:

$t \leftarrow 0$

$N'_0 \leftarrow$ result of deleting edges $\mathcal{E}$ in N

Initialize all edge parameters $PM_0(\hat{U}), SE_0(U)$ (uniformly)

while edge parameters have not converged **do**

  $t \leftarrow t + 1$

  for all deleted edges $U \rightarrow X$ in $\mathcal{E}$ **do**

    $PM_t(\hat{U}{=}u) \leftarrow \alpha \frac{\partial Pr'_{t-1}(\mathbf{e}')}{\partial SE_{t-1}(U=u)}$

    $SE_t(U{=}u) \leftarrow \alpha \frac{\partial Pr'_{t-1}(\mathbf{e}')}{\partial PM_{t-1}(\hat{U}=u)}$

  end for

end while

return N'_t

as those from the previous iteration (up to the first four significant digits). In this case, we shall stop and declare convergence, taking N'_4 as our final approximation N'. Note that this last set of edge parameters are the same as those used in the example given in Section 2.1, which also reports the node marginals given by this ED-BP approximation. To verify the network satisfies the ED-BP properties specified by Condition 1, we have:

$A/\hat{A}$	$Pr'(A \mid \mathbf{e}')$	$Pr'(\hat{A} \mid \mathbf{e}')$	$Pr'(A \mid \mathbf{e}' \setminus \hat{s})$	$Pr'(\hat{A})$
true	0.7135	0.7135	0.8262	0.8262
false	0.2865	0.2865	0.1738	0.1738

3.2 Fully Disconnected Approximations and Polytree Approximations

We show here that ED-BP subsumes, as a degenerate case, the iterative belief propagation algorithm for approximate inference. In particular, the belief propagation algorithm (with messages passed in parallel) corresponds to parametrizing the edges of a simplified network N' that is found by deleting *every* edge in the original network N. We further show that a given belief propagation approximation in fact corresponds to *any* ED-BP approximation as long as the simplified network N' is a polytree.

For an edge $U \rightarrow X$ and an iteration t, let $\pi^t_X(U)$ denote the message passed

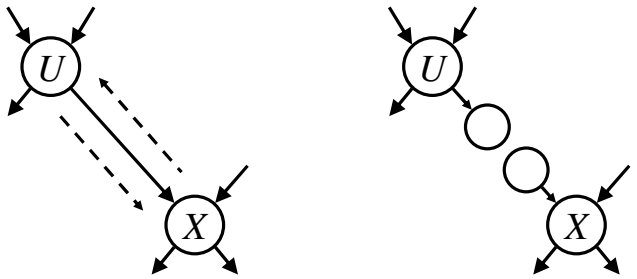


Fig. 4. Correspondence between message passing in IBP and parameter updates in ED-BP.

by IBP from parent U to child X, and let $\lambda_X^t(U)$ denote the message passed from child X to parent U. Moreover, let $b(X|\mathbf{e})$ denote the IBP approximation for the marginal of X. The following theorem then identifies a precise correspondence between:

- the iterations of IBP run on a network N,
- the iterations of ED-BP run on a fully-disconnected approximation N'.

Theorem 2 *Let N be a Bayesian network and N' be the result of deleting every edge from N. Suppose we run IBP on N and* ED-BP *on N', where initially all $\pi_X^0(U) = PM_0(\hat{U})$ and all $\lambda_X^0(U) = SE_0(U)$. Then, for each edge $U \rightarrow X$ and each iteration t, we have:*

- $\pi_X^t(U) = PM_t(\hat{U})$;
- $\lambda_X^t(U) = SE_t(U)$.

Further, for all variables X in N, and for all iterations t, we have

- $b_t(X \mid \mathbf{e}) = Pr'_t(X \mid \mathbf{e}')$.

Figure 4 illustrates this precise relationship between passing messages in IBP and parametrizing edges in ED-BP.

The fully-disconnected case is just a special case of the relationship between IBP and ED-BP. In fact, *any* polytree network N' parametrized by ED-BP corresponds to an IBP approximation for the original network N. Note first that there is a spectrum of approximate networks that are polytrees, ranging from the polytree with no edges (call it the minimal polytree), all the way to polytrees whose edges form a spanning tree of the network variables (call them maximal polytrees). The following theorem relates these different polytrees.

Theorem 3 *Let N be a Bayesian network, and let N' and N'' be two polytrees found by deleting enough edges from N. For every edge parametrization of N' satisfying Condition 1, there is an edge parametrization of N'' also satisfying Condition 1 (and vice versa), where networks N' and N'' agree on node marginals.*

That is, running ED-BP on the minimal polytree network versus a maximal one, or any approximation in between, yields the same node marginals. More precisely, it can be shown that the only difference between these approximate networks is that they lead to different message passing schedules in IBP. In particular, the minimal polytree approximation corresponds to a *parallel* message passing schedule, while maximal polytree approximations correspond to *sequential* schedules (Elidan et al., 2006; Wainwright et al., 2003). Stated more simply, Theorems 2 and 3 tell us the following:

Corollary 1 *Any IBP approximation in a network N is equivalent to the class of polytree* ED-BP *approximations found by deleting edges from N, in the sense that IBP and* ED-BP *agree on their approximations to node marginals.*

Moreover, at any fixed point of IBP (out of potentially multiple fixed points, each potentially having different marginal approximations), there exists a corresponding fixed point for ED-BP in any polytree network N' (and vice versa). Note that previously, (Wainwright et al., 2003) identified an analogous characterization of IBP, as tree-structured approximations embedded in a re-parametrized (undirected) model; we shall discuss the relationship with ED-BP further when we discuss related work in Section 6.

It is important to note that running ED-BP on more connected polytrees can yield more accurate approximations for marginals over multiple nodes. In general, computing such marginals is outside the scope of IBP, yet we will find joint marginals useful in the following section, when we recover deleted edges. We will therefore assume a maximal polytree approximation unless stated otherwise.

Before we proceed, consider, for example, the messages given by IBP for the network given in Figure 3, at convergence:

A	$\pi_B(A)$	$\lambda_B(A)$
true	0.8262	0.3438
false	0.1738	0.6562

A	$\pi_C(A)$	$\lambda_C(A)$
true	0.6769	0.5431
false	0.3231	0.4569

B	$\pi_D(B)$	$\lambda_D(B)$
true	0.7305	0.1622
false	0.2695	0.8378

C	$\pi_D(C)$	$\lambda_D(C)$
true	0.6615	0.4206
false	0.3384	0.5794

In our earlier example, deleting a single edge $A \rightarrow B$ in network N of Figure 3 gives us a polytree network N', and thus, this ED-BP parametrization corresponds to IBP. Note that the IBP messages $\pi_B(A)$ and $\lambda_B(A)$ passed along edge $A \rightarrow B$ are precisely the ED-BP parameters $PM(\hat{A})$ and $SE(A)$ for the

deleted edge $A \rightarrow B$. If we now run the exact belief propagation algorithm in the approximate, polytree network N', we find that messages passed along the remaining edges of N' are given by the corresponding IBP messages above.

3.3 Deciding Which Edges to Recover

Suppose we already have a polytree approximation of the original network, but we are afforded more computational resources. We can then seek to improve the approximation by recovering some of the deleted edges. However, which edge's recovery would have the most positive impact on the quality of the approximation? Alternatively, we can ask: which edge's deletion had the most negative impact in the current approximation? To answer this question, let us consider again Condition 1 which characterizes the edge parameters that ED-BP searches for. Equation 1 of Condition 1 requires that variables U and $\hat{U}$ agree. Equation 2, given Equation 1, says that the soft evidence $\hat{s}$ on variable U should summarize all evidence $\mathbf{e}'$ on its clone $\hat{U}$. This is meant to compensate for an edge deletion which potentially disconnects the evidence now pertaining to $\hat{U}$ which formerly pertained to U. If all such evidence becomes disconnected from U, then Equation 2 is perfectly reasonable. However, if some of the evidence pertaining to $\hat{U}$ remains connected to U in the approximate network, then the soft evidence on U may be overcompensating. One way to measure the extent of this is through the mutual information between U and $\hat{U}$:

$$MI(U; \hat{U} \mid \mathbf{e}') = \sum_{u\hat{u}} Pr'(U=u, \hat{U}=\hat{u} \mid \mathbf{e}') \log \frac{Pr'(U=u, \hat{U}=\hat{u} \mid \mathbf{e}')}{Pr'(U=u \mid \mathbf{e}') Pr'(\hat{U}=\hat{u} \mid \mathbf{e}')}.$$

Note that this mutual information is non-negative, and zero if and only if U is independent of $\hat{U}$. Thus, if deleting $U \rightarrow X$ splits the network into two disconnected subnetworks, this mutual information is zero. Since edge deletion leads to exact results in this case, there is no point in recovering the corresponding edge. On the other hand, if $MI(U; \hat{U}|\mathbf{e}')$ is large, the interaction between U and its clone $\hat{U}$ is strong in the approximate network and, hence, the soft evidence on U may be overcompensating. Edge deletion may have therefore degraded the quality of our approximation, leading us to favor the recovery of such edges.

Algorithm 2 summarizes our proposal for edge recovery. In Steps 1 and 2, we choose an initial network N' based on a maximal polytree embedded in the given network N, and run ED-BP to parametrize the deleted edges. In Step 4, we use the approximate network computed in Step 2 to compute the joint marginals $Pr'(U=u, \hat{U}=u|\mathbf{e}')$ needed for the mutual information scores. We then recover in Step 5 the top k edges with the highest mutual information, and run ED-BP again in Step 6 on a more connected network. We repeat this process, adaptively picking edges to recover, by re-ranking edges

Algorithm 2 ER+A (Adaptive Edge Recovery)

input:

N: a Bayesian network

$\mathbf{e}$: instantiation of some variables in network N

output: approximate network N'

main:

1: $\mathcal{E} \leftarrow$ a set of edges whose deletion renders N a maximal polytree
2: $N' \leftarrow$ ED-BP$(N, \mathbf{e}, \mathcal{E})$
3: **while** recovery of edges in N' is amenable to exact inference **do**
4: rank deleted edges $U \rightarrow X$ based on $MI(U; \hat{U}|\mathbf{e}')$
5: $\mathcal{E} \leftarrow \mathcal{E} - \{$top k edges with the largest scores$\}$
6: $N' \leftarrow$ ED-BP$(N, \mathbf{e}, \mathcal{E})$
7: **end while**
8: **return** N'

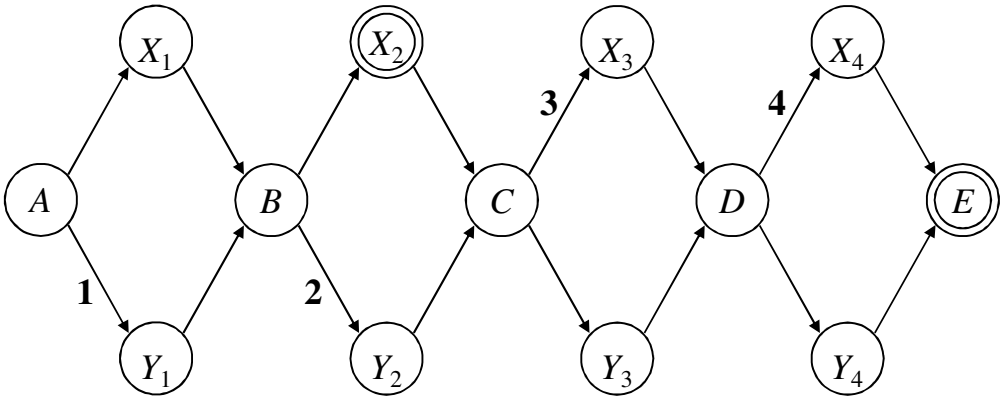
based on the improved approximation. We refer to this method as ER+A, in contrast to ER+S, where we recover edges based only on the initial polytree approximation. This static alternative, can often achieve performance similar to ER+A even though it is less costly to compute. We illustrate this property empirically in Section 5.

Note that our heuristic requires the computation of joint marginals on pairs of variables, which are needed for obtaining the mutual information scores. Consider the fact that

$$Pr'(U=u, \hat{U}=\hat{u}|\mathbf{e}') = Pr'(\hat{U}=\hat{u}|U=u, \mathbf{e}')Pr'(U=u|\mathbf{e}').$$

We can compute all node marginals $Pr'(U=u|\mathbf{e}')$ for all deleted edges in a single run of a polytree or jointree algorithm. We can compute all quantities $Pr(\hat{U}=\hat{u}|U=u, \mathbf{e}')$ for all edges $U \rightarrow X$ outgoing a particular node U by applying exact inference to an approximate network N', once for each state u of U: simply assert each state u as evidence and retrieve the node marginals $Pr'(\hat{U}=\hat{u}|U=u, \mathbf{e}')$. In this manner, Step 4 of our edge recovery algorithm requires that we run an exact algorithm as many times as there are instances u of all edge tails U. If each variable has a constant number of values, and we have n nodes in the network, only $O(n)$ runs of a polytree or jointree algorithm are sufficient to score all deleted edges.

Figure 7 illustrates an example where we recover edges into the network of Figure 6, one edge at a time, until all edges are recovered. Edge recovery is based on the mutual information heuristic as shown in Figure 8. Note that the edge labeled 2 does not improve any of the reported approximations. In fact, if we recovered this edge first, we would see no improvement in approximation quality. This is shown in Figures 9 and 10, where we recover edges with the smallest mutual information first. We see here that the approximations



A	X_1	$\Theta_{X_1\|A}$
true	true	0.75
true	false	0.25
false	true	0.25
false	false	0.75

A	Y_1	$\Theta_{Y_1\|A}$
true	true	0.9
true	false	0.1
false	true	0.1
false	false	0.9

X_1	Y_1	B	$\Theta_{B\|X_1Y_1}$
true	true	true	1.0
true	true	false	0.0
true	false	true	0.1
true	false	false	0.9
false	true	true	0.2
false	true	false	0.8
false	false	true	0.0
false	false	false	1.0

Fig. 5. Variables $X_1, \ldots, X_4$ have the same CPTs, and so do variables $Y_1, \ldots, Y_4$ and variables B, C, D and E. Variable A has $Pr(A{=}\mathsf{true}) = 0.8$. Evidence $\mathbf{e}$ is $X_2{=}\mathsf{false}, E{=}\mathsf{true}$.

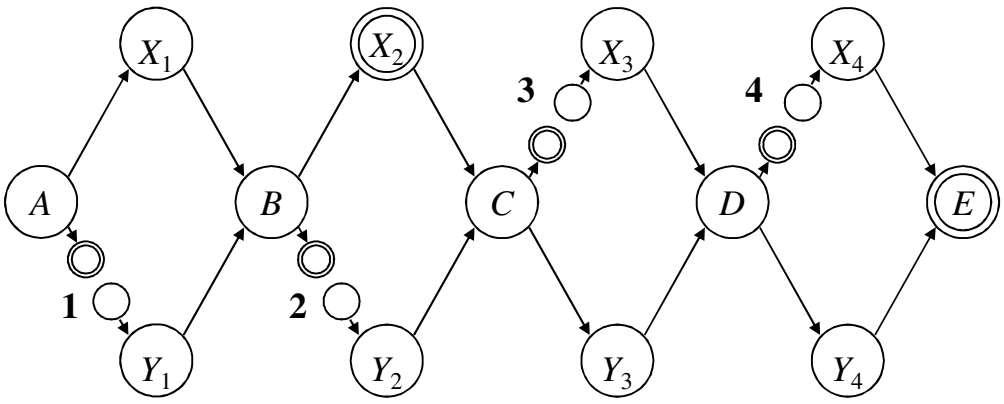


Fig. 6. Deleting four edges in the network of Figure 5.

	Deleted Edges				
Node Marginals	$\{1,2,3,4\}$	$\{2,3,4\}$	$\{2,3\}$	$\{2\}$	$\{\}$
$Pr'(A=\mathsf{true}\|\mathbf{e}')$	0.7114	0.7290	0.7300	0.7331	0.7331
$Pr'(B=\mathsf{true}\|\mathbf{e}')$	0.3585	0.4336	0.4358	0.4429	0.4429
$Pr'(C=\mathsf{true}\|\mathbf{e}')$	0.2405	0.2733	0.2775	0.2910	0.2910
$Pr'(D=\mathsf{true}\|\mathbf{e}')$	0.5824	0.6008	0.6100	0.5917	0.5917

Fig. 7. Improving node marginals by recovering edges into the network of Figure 6. The marginals in the last column are exact.

	Deleted Edges				
Edge Scores	$\{1,2,3,4\}$	$\{2,3,4\}$	$\{2,3\}$	$\{2\}$	$\{\}$
$A \rightarrow Y_1$ (1)	$\mathbf{1.13 \cdot 10^{-3}}$				
$B \rightarrow Y_2$ (2)	0.00	0.00	0.00	**0.00**	
$C \rightarrow X_3$ (3)	$6.12 \cdot 10^{-4}$	$7.26 \cdot 10^{-4}$	$\mathbf{6.98 \cdot 10^{-4}}$		
$D \rightarrow X_4$ (4)	$1.12 \cdot 10^{-3}$	$\mathbf{1.08 \cdot 10^{-3}}$			

Fig. 8. Scoring deleted edges in the network of Figure 6. Scores corresponding to largest mutual information are shown in bold. Based on these scores, edges are recovered according to the order 1, 4, 3 and then 2.

improve more modestly (if at all) than those in Figure 7. This highlights the impact that a good (or poor) recovery heuristic can have on the quality of approximations. We illustrate this point further empirically in Section 5.

Finally, we note that there are a number of alternatives to our mutual information heuristic. One obvious heuristic is to measure the strength of an edge $U \rightarrow X$ as it is defined in the model. That is, we may prefer to delete an edge $U \rightarrow X$ if it is "weak" as defined in the CPT $\Theta_{X|\mathbf{U}}$ (i.e., the conditional probabilities of X do not differ much under changes to the value of U). Note, however, that the mutual heuristic procedure we proposed would leave these edges deleted. If an edge $U \rightarrow X$ is "weak" in N, then the edge $\hat{U} \rightarrow X$ is "weak" in N', and thus the mutual information between the variable U and its clone $\hat{U}$ will be small. Although the notion of "weak" edges is relevant to other more traditional edge deletion approximations (Suermondt, 1992; van Engelen, 1997), because of the auxiliary parameters that we introduce, the quality of an edge deletion approximation depends less on the strength of a particular edge, but more on the degree to which an edge splits a network into two.

We explored an alternative to mutual information as an edge recovery heuristic

	Deleted Edges				
Node Marginals	$\{1,2,3,4\}$	$\{1,3,4\}$	$\{1,4\}$	$\{4\}$	$\{\}$
$Pr'(A=\mathsf{true}\|\mathbf{e}')$	0.7114	0.7114	0.7160	0.7324	0.7331
$Pr'(B=\mathsf{true}\|\mathbf{e}')$	0.3585	0.3585	0.3681	0.4412	0.4429
$Pr'(C=\mathsf{true}\|\mathbf{e}')$	0.2405	0.2405	0.2564	0.2879	0.2910
$Pr'(D=\mathsf{true}\|\mathbf{e}')$	0.5824	0.5824	0.5689	0.5848	0.5917

Fig. 9. Recovering edges into the network of Figure 6. The marginals in the last column are exact.

	Deleted Edges				
Edge Scores	$\{1,2,3,4\}$	$\{1,3,4\}$	$\{1,4\}$	$\{4\}$	$\{\}$
$A \rightarrow Y_1$ (1)	$1.13 \cdot 10^{-3}$	$1.13 \cdot 10^{-3}$	$\mathbf{9.89 \cdot 10^{-4}}$		
$B \rightarrow Y_2$ (2)	**0.00**				
$C \rightarrow X_3$ (3)	$6.12 \cdot 10^{-4}$	$\mathbf{6.12 \cdot 10^{-4}}$			
$D \rightarrow X_4$ (4)	$1.12 \cdot 10^{-3}$	$1.12 \cdot 10^{-3}$	$1.14 \cdot 10^{-3}$	$\mathbf{1.11 \cdot 10^{-3}}$	

Fig. 10. Scoring deleted edges in the network of Figure 6. Scores corresponding to smallest mutual information are shown in bold. Based on these scores, edges are recovered according to the order 2, 3, 1 and then 4.

in (Choi and Darwiche, 2008c), which is based on approximating the mutual information using a soft extension of d-separation. This heuristic can be significantly more efficient in certain cases, yes can be just as effective in finding good edges to recover. We also proposed a more refined mutual information heuristic in (Choi and Darwiche, 2008b) that focuses edge recovery on targeted query variables.

4 Edge Deletion and the Bethe Free Energy

In the previous section, we were concerned primarily with the approximation of node marginals. In this section, we consider approximations to the probability of evidence $Pr(\mathbf{e})$ in a Bayesian network.

In principle, we can use the exact probability of evidence computed in an approximate network as an approximation to the true value of $Pr(\mathbf{e})$. Unfortunately, in an ED-BP approximation, the probability of evidence $Pr'(\mathbf{e}')$ by itself does not lead to good results. Luckily, it is possible to rectify these approximations in a way that leads to exact results in certain situations. As

the previous section led to a new perspective on iterative belief propagation, this section brings us a new perspective on the intimately related Bethe free energy approximation (Yedidia et al., 2005).

Consider again the case where we delete a single edge $U \rightarrow X$, but where the mutual information $MI(U; \hat{U}|\mathbf{e}')$ is zero in the resulting network. Let us call such an edge a zero-MI edge. Remember that an edge $U \rightarrow X$ that splits a network into two disconnected subnetworks is also a zero-MI edge, and in this case, Lemma 1 tells us the subnetwork marginals are exact when we use ED-BP edge parameters. The following proposition tells us that, in the more general situation where a single zero-MI edge is deleted, we can also recover the true probability of evidence.

Proposition 1 *Let N be a Bayesian network, and let N' be the result of deleting a single edge $U \rightarrow X$ in N. Suppose further that the edge parameters of network N' satisfy Condition 1 and that $MI(U; \hat{U}|\mathbf{e}') = 0$ in N'. Then*

$$Pr(\mathbf{e}) = Pr'(\mathbf{e}') \cdot \frac{1}{z_{UX}} \quad \textit{where} \quad z_{UX} = \sum_u PM(\hat{U}{=}u)SE(U{=}u).$$

That is, if we delete a single edge $U \rightarrow X$ and find that U and X are independent in the approximate network N', we can correct the approximate probability of evidence $Pr'(\mathbf{e}')$ by z_{UX} and recover the exact value of $Pr(\mathbf{e})$.[4] Note that since $z_{UX} \leq 1$ (since we assume $PM(\hat{U})$ and $SE(U)$ normalize to sum to one), this correction can only increase the value of $Pr'(\mathbf{e}')$.

This observation leads to an approximation of the probability of evidence, that is exact in the case a single zero-MI edge is deleted. Note that the correction suggested by Proposition 1 is specified purely in quantities available in the approximate network N', which may become computable when we delete enough edges. From this perspective, Proposition 1 suggests an *approximate* correction scheme when deleting multiple edges $U \rightarrow X$, where we use a multiplicative accumulation of the corrections for each edge. In particular, Proposition 1 suggests the following correction to the probability of evidence, $Pr'(\mathbf{e}') \cdot \frac{1}{z}$, where

$$z = \prod_{U \rightarrow X} z_{UX} = \prod_{U \rightarrow X} \sum_u PM(\hat{U}{=}u)SE(U{=}u). \tag{7}$$

In general, this correction yields only approximate results, but as we shall soon see, this correction leads to a new perspective on the Bethe free energy.

[4] It is also possible to show, for the deletion of a single zero-MI edge, that the marginals $Pr'(U{=}u|\mathbf{e}') = Pr'(\hat{U}{=}u|\mathbf{e}')$ given by ED-BP are in fact the exact marginals $Pr(U{=}u|\mathbf{e})$. However, the marginals of other variables may not be exact.

4.1 The Bethe Free Energy and the Approximate Probability of Evidence

In (Yedidia et al., 2005), a fixed point of iterative belief propagation was shown to correspond precisely to stationary points of the Bethe free energy, which we review below. This connection was a significant step towards understanding belief propagation, and led to the highly influential generalized belief propagation (GBP) algorithms (Yedidia et al., 2005); see Section 6 for a description of GBP. In this section, we show that the Bethe free energy approximation is in fact the exact probability of evidence (suitably corrected) of a simplified Bayesian network found by deleting edges.

Consider first the KL–divergence between two distributions Pr and Pr' given that each has been conditioned on evidence $\mathbf{e}$:

$$KL(Pr'(. \mid \mathbf{e}), Pr(. \mid \mathbf{e})) = \sum_{w} Pr'(w \mid \mathbf{e}) \log \frac{Pr'(w \mid \mathbf{e})}{Pr(w \mid \mathbf{e})}. \tag{8}$$

The KL–divergence is non-negative and equal to zero if and only if $Pr'(.|\mathbf{e})$ and $Pr(.|\mathbf{e})$ are equivalent. However, the KL–divergence is not a true distance measure in that it is not symmetric.

Note that the KL–divergence can be written as a sum of three components:

$$KL(Pr'(. \mid \mathbf{e}), Pr(. \mid \mathbf{e})) = U - H + \log Pr(\mathbf{e})$$

where

- energy $U = -\sum_{X\mathbf{U}} E[\log \Theta_{X|\mathbf{U}} \lambda_X] = -\sum_{X\mathbf{U}} \sum_{x\mathbf{u}} Pr'(x\mathbf{u}|\mathbf{e}) \log \theta_{x|\mathbf{u}} \lambda_x$ is a set of expectations over the original network parameters weighted by the approximate distribution. Here $\theta_{x|\mathbf{u}}$ is a network CPT, and λ_x is an evidence indicator, where $\lambda_{\mathbf{e}}(x) = 1$ if x is consistent with evidence $\mathbf{e}$ and zero otherwise.
- entropy $H = -\sum_w Pr'(w|\mathbf{e}) \log Pr'(w|\mathbf{e})$ is a measure of the uncertainty associated with the approximate distribution $Pr'(.|\mathbf{e})$.

See (Cover and Thomas, 1991) for more on entropy and the KL–divergence.

The KL–divergence suggests a formulation of approximate inference as an optimization problem. In particular, we can pick a parametrized form of an approximate distribution $Pr'(.|\mathbf{e})$, and search for an instance of it that (hopefully) minimizes the KL–divergence. Since the $\log Pr(\mathbf{e})$ term is independent of the choice of an approximating distribution, we can equivalently minimize a free energy $F = U - H$. This formulation thus reveals two competing properties of $Pr'(.|\mathbf{e})$ that minimize a free energy F, and hence, the KL–divergence:

- $Pr'(.|\mathbf{e})$ should match the original distribution by giving more weight to

more likely parameters $\theta_{x|\mathbf{u}}$ (i.e, minimize the energy U);
- $Pr'(.|\mathbf{e})$ should not favor unnecessarily one instantiation over another by being evenly distributed (i.e., maximize the entropy H).

Note that in the case where $Pr'(.|\mathbf{e})$ and $Pr(.|\mathbf{e})$ are equivalent, and the KL–divergence is zero, we have that $F = -\log Pr(\mathbf{e})$.

In (Yedidia et al., 2005), iterative belief propagation was identified to be an approximation as described above, where the approximating distribution is assumed to factorize as follows:

$$Pr'(. \mid \mathbf{e}) = \prod_{X\mathbf{U}} \frac{Pr'(X\mathbf{U} \mid \mathbf{e})}{\prod_{U \in \mathbf{U}} Pr'(U \mid \mathbf{e})}, \tag{9}$$

Note how this form is in terms of only the marginals that Pr' assigns to parents and families. A number of observations are in order about this assumption. First, this choice of $Pr'(.|\mathbf{e})$ is expressive enough to describe distributions $Pr(.|\mathbf{e})$ induced by polytree networks N. That is, if N is a polytree, then the corresponding distribution $Pr(.|\mathbf{e})$ does indeed factorize according to Equation 9. In the case where N is not a polytree, then we are simply trying to fit $Pr(.|\mathbf{e})$ into an approximation $Pr'(.|\mathbf{e})$ as if it were generated by a polytree network.

This choice of distribution specifies a particular approximation to the free energy $F = U - H$, known as the Bethe free energy:

$$F_\beta = U_\beta - H_\beta$$

where U_β is the Bethe *average* energy

$$U_\beta = -\sum_{X\mathbf{U}} E[\log \Theta_{X|\mathbf{U}} \lambda_X] \tag{10}$$

and H_β is the Bethe *approximate* entropy

$$H_\beta = \sum_{X\mathbf{U}} H(X\mathbf{U} \mid \mathbf{e}) - \sum_{X\mathbf{U}} \sum_{U \in \mathbf{U}} H(U \mid \mathbf{e}), \tag{11}$$

where our expectations and entropies are now based on marginal approximations $b(x\mathbf{u}|\mathbf{e})$ and $b(u|\mathbf{e})$ from IBP:

- $E[\log \Theta_{X|\mathbf{U}} \lambda_X] = \sum_{x\mathbf{u}} b(x\mathbf{u} \mid \mathbf{e}) \log \theta_{x|\mathbf{u}} \lambda_x$
- $H(X\mathbf{U}|\mathbf{e}) = -\sum_{x\mathbf{u}} b(x\mathbf{u} \mid \mathbf{e}) \log b(x\mathbf{u} \mid \mathbf{e})$
- $H(U|\mathbf{e}) = -\sum_{u} b(u \mid \mathbf{e}) \log b(u \mid \mathbf{e})$

Under certain consistency and normalization constraints, one can show that family marginals $b(x\mathbf{u}|\mathbf{e})$ and node marginals $b(u|\mathbf{e})$ are stationary points of the Bethe free energy if and only if IBP messages are a fixed point of iterative belief propagation.

As we stated earlier, when a network is a polytree, the distribution $Pr(.|\mathbf{e})$ does indeed factorize according to Equation 9. Thus, the corresponding KL–divergence is zero, and we have that F_β is exactly $-\log Pr(\mathbf{e})$. In principle, we can simply use F_β as an approximation to the probability of evidence, even when a network is not a polytree, i.e., $Pr_\beta(\mathbf{e}) = \exp\{-F_\beta\}$. As the following theorem asserts, this approximation is precisely an edge-corrected probability of evidence, as suggested by Proposition 1.

Theorem 4 *Let N be a Bayesian network, and let N' be the result of deleting edges $U \rightarrow X$ in N. Suppose further that the edge parameters of network N' satisfy Condition 1, and that N' is rendered a polytree. Then*

$$Pr_\beta(\mathbf{e}) = Pr'(\mathbf{e}') \cdot \frac{1}{z} \quad \textit{where} \quad z = \prod_{U \rightarrow X} z_{UX} = \prod_{U \rightarrow X} \sum_u PM(\hat{U}{=}u) SE(U{=}u).$$

That is, the Bethe approximation of the probability of evidence is actually the corrected probability of evidence of an approximate ED-BP network, but interestingly, for any polytree approximation N'. Moreover, when the original network N happens to be a polytree, the correction yields the exact value of $Pr(\mathbf{e})$. In general, when N has (undirected) cycles, the correction of the probability of evidence is approximate and leads to a class of region-based free energy approximations when an arbitrary number of edges are deleted (Yedidia et al., 2005); see Appendix D for the relationship between ED-BP and iterative joingraph propagation approximations.

The following proposition parallels the correspondence between fixed points of IBP and stationary points of the Bethe free energy (Yedidia et al., 2005).

Proposition 2 *Let N be a Bayesian network, and let N' be the result of deleting edges $U \rightarrow X$ in N. Edge parameters of N' are an* ED-BP *fixed point if and only if the edge parameters are a stationary point of $Pr'(\mathbf{e}') \cdot \frac{1}{z}$.*

This proposition tells us that an IBP fixed point that is stationary in the Bethe free energy F_β is an ED-BP fixed point that is stationary in an edge-corrected probability of evidence. Note again that the factor z is the correction factor for the correction scheme based on Proposition 1, which is exact in the case where a single zero-MI edge is deleted.

4.2 An Example

Consider again our example from Section 2.1, where we delete a single edge $A \rightarrow B$, using ED-BP edge parameters:

$\hat{S}$	A	$\Theta_{\hat{S}\|A} = SE(A)$
true	true	0.3438
true	false	0.6562

$\hat{A}$	$\Theta_{\hat{A}} = PM(\hat{A})$
true	0.8262
false	0.1738

Here, the probability of evidence in the approximate network is $Pr'(\mathbf{e}') \approx 0.1446$, whereas the true probability of evidence is $Pr(\mathbf{e}) = 0.38$. In this example, $A \rightarrow B$ is not a zero-MI edge, so the factor $z_{AB} \approx 0.3438 \cdot 0.8262 + 0.6562 \cdot 0.1738 = 0.3981$ does not provide an exact correction. However, $Pr'(\mathbf{e}') \cdot \frac{1}{z_{AB}} \approx 0.3633$ provides a better approximation to $Pr(\mathbf{e})$ than $Pr'(\mathbf{e}')$ does alone.

We can also compute the Bethe average energy $U_\beta \approx 2.648$ and the Bethe approximate entropy $H_\beta \approx 1.635$, giving us for the Bethe free energy $F_\beta = U_\beta - H_\beta \approx 1.013$ (using natural logarithms). The Bethe approximation to the probability of evidence is $Pr_\beta(\mathbf{e}) = \exp\{-F_\beta\} \approx 0.3633$, which is indeed the probability of evidence $Pr'(\mathbf{e}')$ of our approximate network, after applying a correcting factor z_{AB}.

Suppose now for the network given in Figure 3, that we replace the CPT for D with the following:

B	C	D	$\Theta_{D\|BC}$
true	true	true	0.162
true	false	true	0.81
false	true	true	0.18
false	false	true	0.9

(omitting the redundant rows for D=false). Still conditioning on evidence $\mathbf{e} : D$=true, we find that this time edge $A \rightarrow B$ is in fact a zero-MI edge, when we use the ED-BP edge parameters:

$\hat{S}$	A	$\Theta_{\hat{S}\|A} = SE(A)$
true	true	0.4894
true	false	0.5106

$\hat{A}$	$\Theta_{\hat{A}} = PM(\hat{A})$
true	0.9231
false	0.0769

The probability of evidence in the approximate network is $Pr'(\mathbf{e}') \approx 0.2121$ and our correction factor is $z_{AB} \approx 0.4894 \cdot 0.9231 + 0.5106 \cdot 0.0769 = 0.4910$. Our corrected probability of evidence is $Pr'(\mathbf{e}') \cdot \frac{1}{z_{AB}} = 0.432$ which is now exactly the true probability of evidence $Pr(\mathbf{e})$.

The Bethe average energy is $U_\beta \approx 2.067$ and the Bethe approximate entropy is $H_\beta \approx 1.228$, giving us the Bethe free energy $F_\beta = U_\beta - H_\beta \approx 0.8393$. Since our corrected probability of evidence is exact, the Bethe approximation yields the exact result $Pr_\beta(\mathbf{e}) = \exp\{-F_\beta\} = 0.432$.

4.3 Deciding Which Edges to Recover

In Section 3.3, we proposed a mutual information heuristic for improving marginal approximations that is based on recovering first those edges with a large mutual information. Since Proposition 1 tells us that a corrected probability of evidence is exact when a single zero-MI edge is deleted, we may expect the same mutual information heuristic to be effective for improving approximations to the probability of evidence as well. Indeed, we shall show empirically that this is the case in the following section.

5 Experiments

In this section, we provide experimental comparisons of the approximations given by ED-BP, IBP, and the Bethe free energy. Our goal is to highlight the impact that good (and bad) heuristics for edge recovery can have on the quality of the resulting approximations.

The experiments in this section were performed on a 2.83GHz Intel Xeon Processor (E5440). [5]

5.1 Edge Recovery and Marginal Approximations

We start by observing the degree to which ED-BP and edge recovery is capable of improving the quality of marginal approximations given by IBP, over a selection of real-world and random networks. [6]

[5] An inference system based on ED-BP participated in the Probabilistic Inference Evaluation at UAI'08 (Darwiche et al., 2008), which included other systems based on belief propagation and its generalizations. Note, however, that some of these systems, including the ones based on ED-BP and IJGP, did use other techniques, such as preprocessing—including, for example, the use of SAT engines. Hence, it is not obvious how to distinguish the merits of the different variants of BP, as opposed to those based on the preprocessing used.

[6] Many of the networks used here in the empirical analysis are available at `http://www.cs.huji.ac.il/labs/compbio/Repository/`. Networks `emdec` and

For each network, we generated at least 100 evidence instantiations, found by setting all leaf nodes according to their joint distribution (except for networks `emdec` and `tcc`, where we set values on leaves at random as the joint distribution is highly skewed). For each evidence instantiation, we averaged the marginal errors over all unobserved variables, where the error of a marginal approximation for a variable X is the KL–divergence $KL(Pr(X|\mathbf{e}), Pr'(X|\mathbf{e}'))$ between the exact and approximate marginals. We considered only instances where IBP converged within 200 iterations (less than 3 instances per 100 networks did not converge), where convergence for IBP (and ED-BP) was determined when every message (or parameter) of an iteration is within 10^{-8} of the previous.

Consider Figure 11, which depicts the improved approximations for a variety of Bayesian networks, comparing IBP and our edge recovery algorithm. The x–axes of these figures correspond to the number of edges recovered, starting from no edges recovered (a polytree approximation/IBP) to all edges recovered (original network/exact inference). We determined initial polytrees randomly, and recovered $\frac{1}{10}$-th of the deleted edges at a time. We also used a jointree algorithm for exact inference in ED-BP. We compared edge recovery with adaptive rankings (ER+A), static rankings (ER+S), and random recovery. Further, we reversed the rankings (recovering edges with the lowest mutual information) for the non-random cases (ER-A, ER-S).

In Figure 11, we see that in the `alarm` and `pigs` networks, significant improvements can be seen in recovery by ER+A and ER+S over random recovery, even when a small number of edges are recovered. In networks `water`, `noisy-or` (described later in the section), `tcc` and `emdec`, we observed that ER+A yields better approximations than ER+S, after recovering the first set of edges, and that ER+S typically yields better approximations than random recovery. In networks `hailfinder` and `pigs`, we see that static recovery can perform as well as adaptive recovery for certain networks. In network `win95pts`, ER+S happened to perform better than ER+A. In all networks, we see heuristics ER-A and ER-S can similarly identify edges that do not have as much impact on the approximation. In fact, we see that a substantial number of edges can be recovered without much benefit, highlighting the importance of a good edge recovery heuristic.

Consider Figure 12, which examines the computational impact edge recovery has in the `pigs` network. First, we compare the effect that edge recovery has on approximation quality (as in Figure 11) and the size of the resulting jointree. Here, the size of a jointree is measured as the number of entries in cluster tables, summed over all clusters, then taking the $\log_2$. This roughly estimates the complexity of a Hugin-based jointree algorithm. We see that recovery by

`tcc` are noisy-or networks for diagnosis, courtesy of HRL Laboratories, LLC.

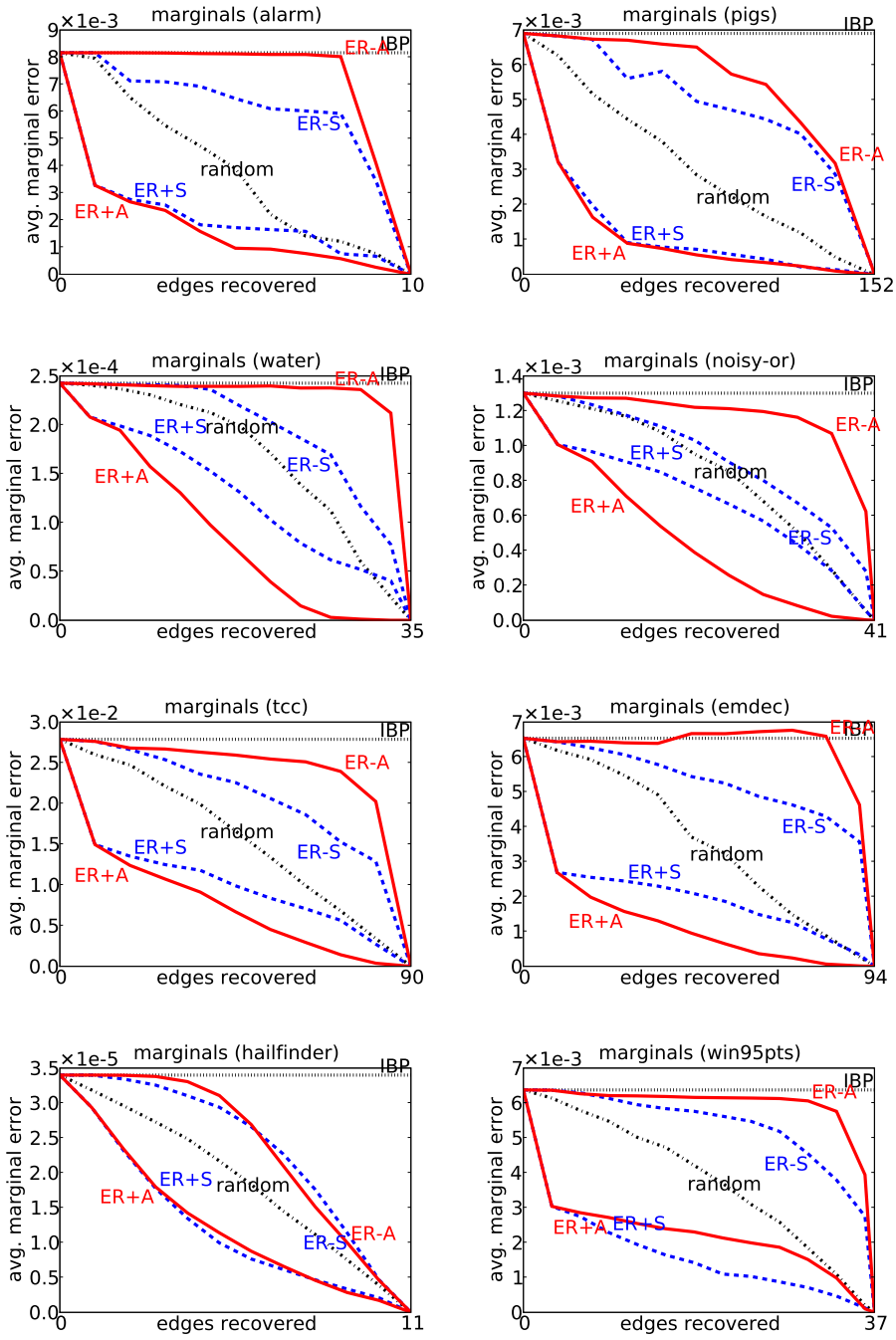


Fig. 11. Quality of approximation vs edges recovered.

ER+A happens to result in jointrees that are modestly more compact than those found by random edge recovery. On the other hand, ER-A results in more computationally demanding jointrees. More importantly, we see that as we start to recover edges, we can indeed identify better approximations, without impacting much the computational complexity of inference. We have made similar observations for other networks as well, including the `barley` network in Figure 13.

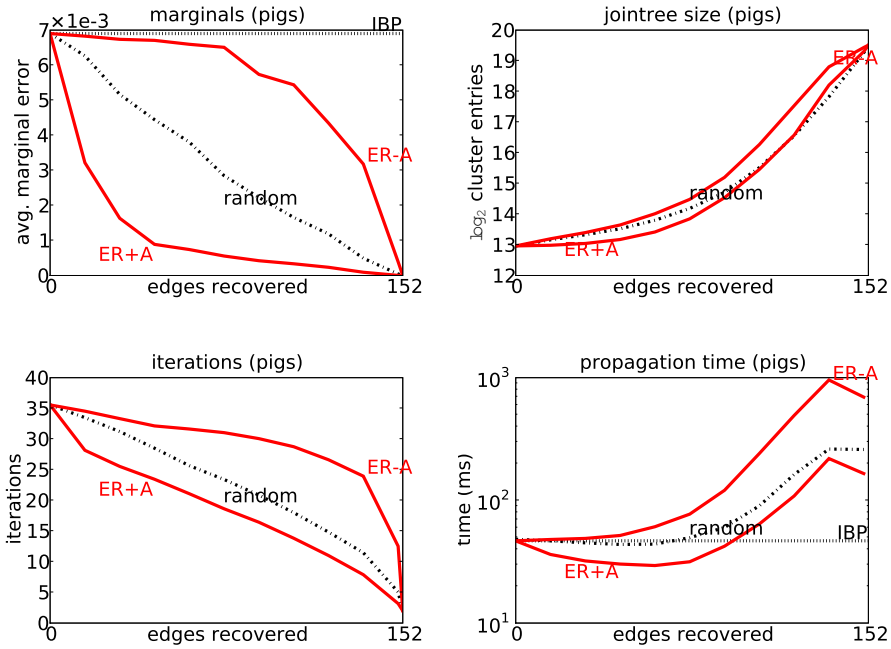


Fig. 12. Effect of edge recovery on quality of approximation, complexity of iteration, convergence rate, and running time.

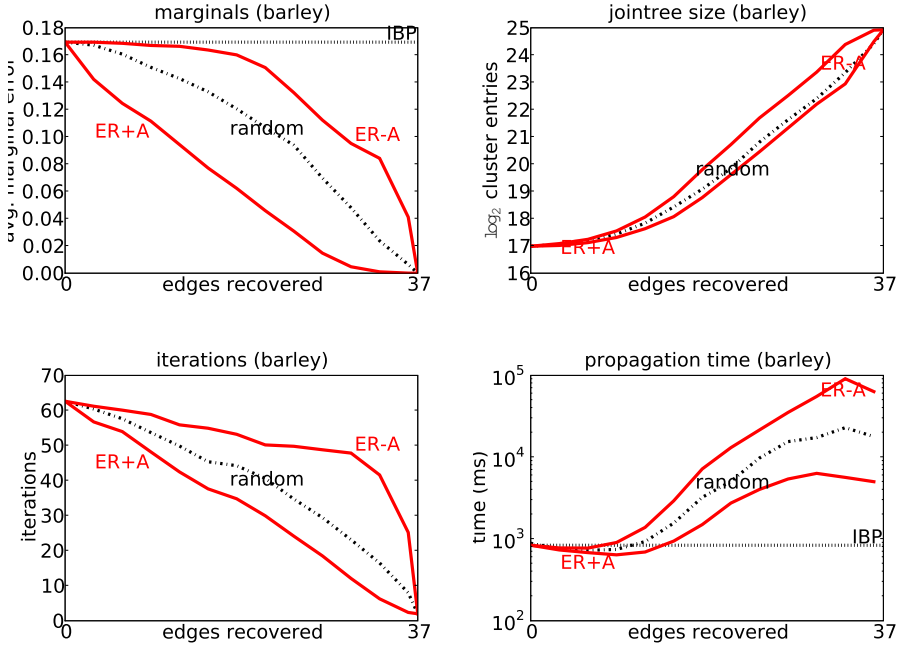


Fig. 13. Effect of edge recovery on quality of approximation, complexity of iteration, convergence rate, and running time.

It is important to note here that when recovering edges, we may increase network complexity, but we may also reduce convergence time. Hence, one must factor both issues when evaluating the costs incurred when recovering edges. In particular, we see in the second row of Figure 12 that ER+A can improve the

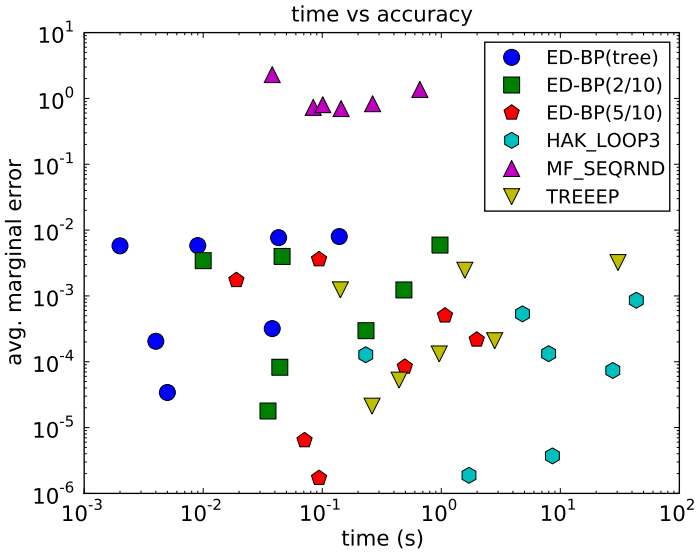


Fig. 14. Quality of approximation vs computation time. We compared ED-BP with an initial tree approximation, ED-BP with adaptive edge recovery of $\frac{2}{10}$ and $\frac{5}{10}$ of the edges, GBP (HAK_LOOP3), mean-field (MF_SEQRND), and Tree-EP (TREEEP).

rate of convergence as more edges are recovered. Combined with only modest gains in the complexity of inference, the overall time to compute an ED-BP approximation with edges recovered can in fact be computationally more efficient than ED-BP in a polytree with no edges recovered, and correspondingly, IBP. Indeed, we observe that this is the case in the fourth plot of Figure 12.

Finally, in Figure 14, we compare time and accuracy trade-offs for a number of approximations:

- ED-BP (tree): an ED-BP approximation using a random spanning tree (which corresponds to IBP);
- ED-BP ($\frac{2}{10}$, $\frac{5}{10}$): adaptive edge recovery (ER+A) starting with a random spanning-tree, recovering $\frac{2}{10}$-ths and $\frac{5}{10}$-ths of the model edges, recovering $\frac{1}{10}$-th of the model edges at a time;
- HAK_LOOP3: a convergent generalized belief propagation (GBP) algorithm (Heskes et al., 2003; Yedidia et al., 2005);
- MF_SEQRND: a mean-field algorithm;
- TREEEP: a Tree-EP algorithm (Minka and Qi, 2003).

We describe GBP, mean-field and EP in Section 6. The implementations we used come from the freely available libDAI library Mooij (2008); HAK_LOOP3, MF_SEQRND and TREEEP are the aliases of the algorithms used from this library. Note that our implementation of ED-BP and ER+A is in Java, whereas the libDAI library is implemented in C++. In general, we expect C++ to be more computationally efficient than Java. We ran each algorithm with a limit of 200 iterations, with a convergence threshold of 10^{-8}. Note that the running

times we report for edge recovery accumulate the time to rank edges and to re-run ED-BP.

We evaluated these algorithms over the networks `alarm`, `emdec`, `hailfinder`, `pathfinder`, `pigs`, `water`, and `win95pts`. Each plot point in Figure 14 represents a network and algorithm, averaged over 200 evidence instantiations (which were generated in the same way we described previously). We see that the mean field approximation MF_SEQRND is less accurate, and it is the only algorithm we evaluated that does not generalize belief propagation. Of the remaining approximations, we find that ED-BP (tree) tends to be the most efficient, but tends to have less accurate approximations. As expected, we find that recovering edges via ER+A tends to improve the approximation at the expense of longer running times. HAK_LOOP3 and TREEEP, both also generalizations of belief propagation, improve on the accuracy of ED-BP (tree), but we find that in these networks, they tend to be more computationally expensive to compute than ER+A, even when half of the model edges are recovered. Note that the algorithm HAK_LOOP3 that we evaluated is a convergent, double-loop algorithm, which are typically less efficient, although we found the corresponding single-loop implementation GBP_LOOP3 to exhibit unsatisfactory convergence behaviour in the benchmarks we evaluated.

5.2 Edge Recovery and Approximations to the Probability of Evidence

We evaluate now the effectiveness of edge recovery in improving approximations to the probability of evidence, as compared to the Bethe free energy approximation. Here we measure the quality of an approximation in terms of the relative error in the log probability of evidence: $|\log Pr(\mathbf{e}) - \log Pr'(\mathbf{e}') \cdot \frac{1}{z}| / \log Pr(\mathbf{e})$. Again, we start with a random spanning tree approximation of the original network, which now corresponds to the Bethe free energy approximation, and recover edges until all edges are recovered.

In Figure 15, we see that our edge recovery heuristics have a similar impact on the quality of a $Pr(\mathbf{e})$ approximation as they do on marginals approximations (in Figure 11). When no edges are recovered, our approximation is a polytree, and we have the Bethe approximation $Pr_\beta(\mathbf{e}) = \exp\{-F_\beta\}$ of the probability of evidence. As we start to recover, we see that ER+A and ER+S can be considerably more effective than edge recovery, in terms of improving an approximation. We further note that a poor choice in edge recovery heuristics can in fact lead to a worse approximation than that given by the Bethe free energy.

Noisy-or networks. We conclude this section by examining more closely the set of experiments on random two-layer noisy-or networks. We generated over 100

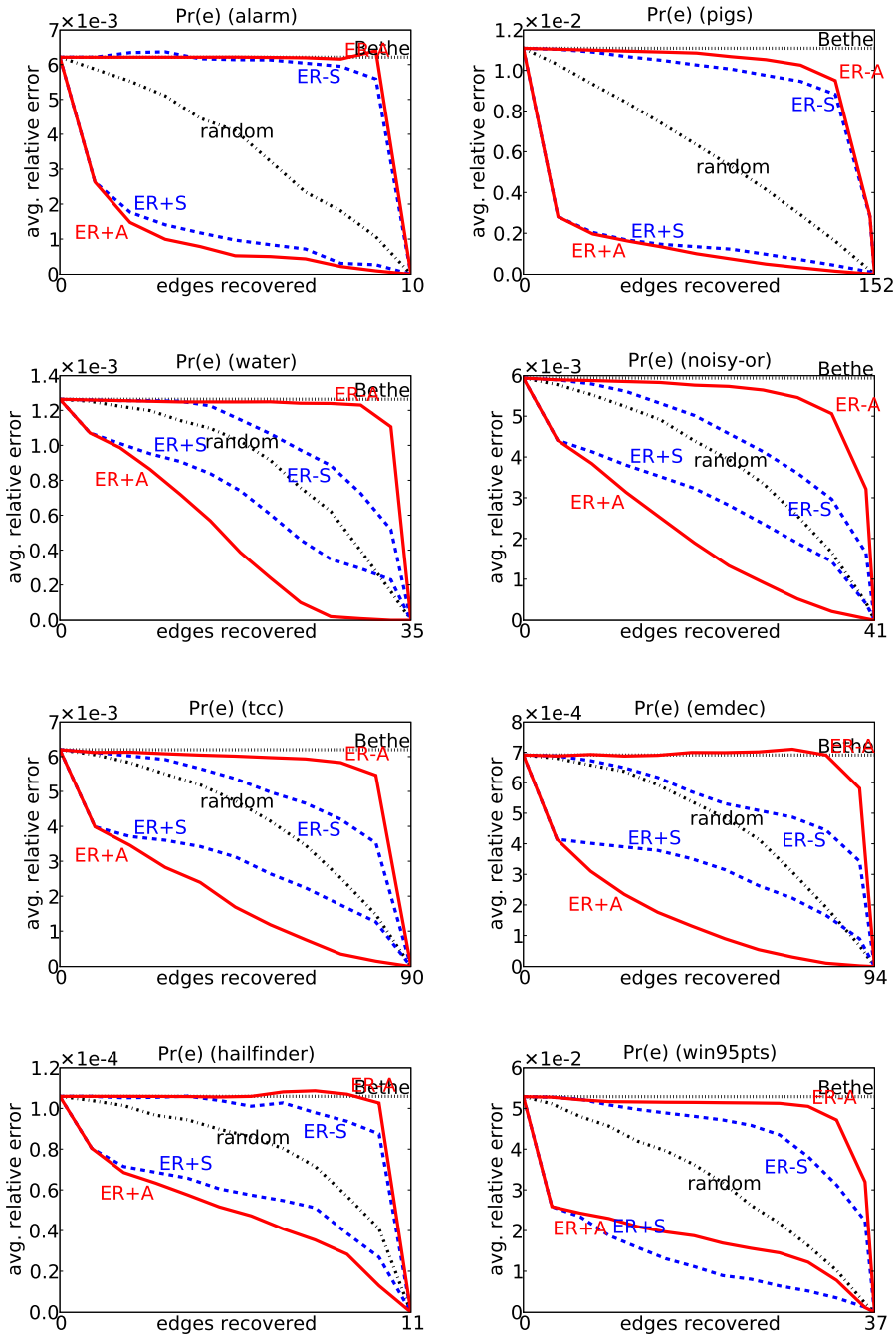


Fig. 15. Quality of approximation vs edges recovered.

randomly parametrized noisy-or networks, each with 20 roots and 20 sinks, where sinks are given 4 random parents. For each network, we chose some random number k between 0 and 20, and observed k sinks as positive findings and the remaining sinks as negative findings. Priors for the roots, and suppressor probabilities for the sinks, were also generated randomly.

Deleting an edge in a two-layer noisy-or network effectively disconnects a cause

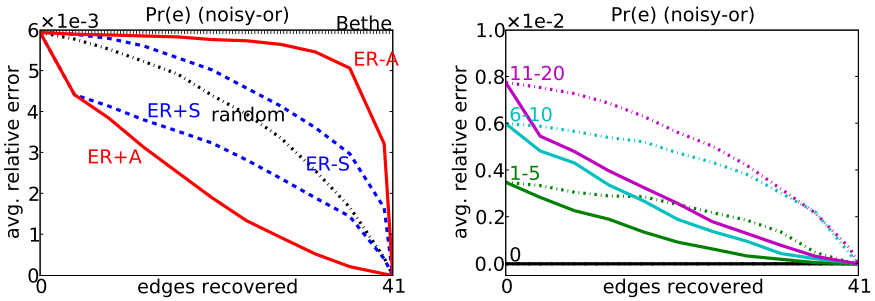


Fig. 16. Quality of approximation vs edges recovered in noisy-or networks.

variable C from an effect variable E, where a clone $\hat{C}$ replaces C as a a cause of E. In this situation, we may appeal to the semantics of edge deletion to reason how well ED-BP and the Bethe approximation may perform. With no positive findings, for example, we know that all causes are pairwise mutually independent, including a cause C and its clone $\hat{C}$ in a noisy-or network where edges have been deleted. Starting from a tree-structured approximation (corresponding to the Bethe approximation), every recoverable edge is zero-MI, and will remain zero-MI up to the point where all edges are recovered. Thus we may infer that all corrections to the probability of evidence to be exact throughout, and thus also that the Bethe approximation is exact.

Consider Figure 16, which compares the quality of our approximations on the probability of evidence. On the left, we plot the overall behavior, just as in Figure 15. On the right of Figure 16, we observe how the quality of an approximation varies with the number of positive findings in a noisy-or network. We have 4 pairs of plots: the dotted lines represent random recovery, and the solid lines represent recovery using ER+A. Each pair corresponds to averages over a range of positive findings: 0, 1–5, 6–10, and 11–20. With no positive findings, we see that ED-BP yields exact results, as predicted, even when no edges are recovered. Given this, it is not unexpected to see that as the number of positive findings increase the quality of the approximations degrade as there are fewer and fewer zero-MI edges.

6 Related Work

Iterative belief propagation has received a good deal of interest over the years as an approximate inference algorithm, particularly since revolutionary decoders for error correction have been shown to be instances of this algorithm (Frey and MacKay, 1997; McEliece et al., 1998). A number of formulations, generalizations, and alternatives to IBP have since been proposed.

In this paper, we proposed a framework for approximate inference based on deleting edges that subsumes, as a degenerate case, this influential algorithm.

In particular, we have shown that

- IBP marginal approximations are the exact marginals of a simplified network found by deleting edges; and
- the Bethe approximation to the probability of evidence is the exact probability of evidence (suitably corrected) in the same simplified network.

In particular, we found that belief propagation approximations correspond to simplified networks where:

- *every* edge is deleted from the original model;
- but also where enough edges have been deleted from the original model to render the model a polytree, *any* polytree.

Some of these properties are evident, or analogous to, properties of belief propagation visible from other, albeit disparate, perspectives. For example, IBP marginal approximations also arise from a fully-disconnected expectation propagation approximation (Minka, 2001). They have also been characterized in terms of trees embedded in network *re*-parametrizations (Wainwright et al., 2003). As we highlighted in Section 4, belief propagation also arises from free-energy approximations based on tree-based factorizations of probability distributions (Yedidia et al., 2005).

Considering that these observations were previously made from what were seemingly different perspectives, we argue that an edge deletion perspective is a particularly natural one for analyzing and understanding belief propagation. For example, an edge deletion semantics allowed us to propose a new, but simple, edge recovery algorithm for effectively finding approximations more structured and more accurate than that of iterative belief propagation. Indeed, in other generalizations of belief propagation, it is a non-trivial task to identify, for a given model, the form that a good approximation should take.

Generalizations of belief propagation.

Among generalizations of IBP, generalized belief propagation (GBP) algorithms are one of the most notable (Yedidia et al., 2005), and perhaps one of the most general (Welling et al., 2005; Wainwright et al., 2003). Whereas IBP passes messages along edges in a network N, GBP algorithms pass messages according to an auxiliary structure composed of regions of N, where larger regions provide more accurate approximations. ED-BP can in fact be simulated by a particular class of GBP algorithms based on joingraphs (Aji and McEliece, 2001; Dechter et al., 2002); for details, see Appendix D. Therefore, our method identifies a new subclass of GBP, where our edge recovery procedure can be seen as an approach to identifying good GBP approximations, and in particular, good joingraphs.

Expectation propagation (EP) algorithms are another class of algorithms where belief propagation falls as a special case (Minka, 2001; Welling et al., 2005). EP is based on assuming a simplified model as a base approximation, which is iteratively fitted with other more structured approximations. Iterative belief propagation arises as a special case of expectation propagation in the case where the base approximation is fully-disconnected, and we try to fit into it models where individual families are recovered. EP and ED-BP both correspond to IBP in the fully-disconnected case (where every edge is deleted, in the case of ED-BP). Unlike ED-BP, the tree-structured nature of IBP approximations is not as evident in EP, as an EP approximation based on a tree will incorporate higher-order structure, and will typically begin to provide improved approximations (Minka and Qi, 2003).

We can see the tree-structured nature of IBP in tree-based re-parametrizations (TRP) (Wainwright et al., 2003). This algorithm *re*-parametrizes a model, also in an iterative fashion, by re-factorizing trees embedded in the network. The original distribution is left invariant to this iterative process, but interestingly, exact inference in any tree-structured subnetwork of the model yields the same marginal approximations given by IBP.

Among these generalizations of IBP, the design of approximating structures has received considerably less attention. In GBP, one needs to provide a region graph, which specifies (roughly) which regions of the original network should be handled exactly, and to what extent different regions should be mutually consistent. Similarly, in TRP, one needs to specify a hypergraph or joingraph on which a re-parametrization is performed. In EP, one needs to specify an EP-graph, which organizes a set of approximate models around a base approximation.

The only proposal that the authors are aware of for the systematic design of approximations in this class are region pursuit algorithms (Welling, 2004; Welling et al., 2005), which start with an IBP approximation and looks for good candidates among a set of proposed regions, using local message propagations to evaluate those that should be added to the approximation. In particular, regions are added when they lead to the greatest change in the distribution. This approach relies in part on the assumption that a more structured approximation will lead to an improved approximation. However, one still needs a number of heuristics to help avoid those regions that will in fact lead to worse approximations (Yedidia et al., 2005; Welling et al., 2005).

Variational approximations.

Variational methods are another class of algorithms that formulate approximate inference as the problem of exact inference in some simplified model, often seeking to minimize the KL–divergence (e.g., Jordan et al., 1999; Jaakkola,

2001; Wiegerinck, 2000; Geiger et al., 2006; Choi and Darwiche, 2006b). One example is the mean-field method, where we seek to approximate a network N by a fully-disconnected approximation N' (Haft et al., 1999). Other variational approaches typically assume particular structures in their approximate models, such as chains (Ghahramani and Jordan, 1997), trees (Frey et al., 2000; Minka and Qi, 2003), or disconnected subnetworks (Saul and Jordan, 1995; Xing et al., 2003).

Although IBP can also be viewed from a variational perspective (Yedidia et al., 2005), mean-field and related methods are distinct in that such approximations guarantee lower bounds on the probability of evidence, whereas the Bethe approximation (and the edge-corrected probability of evidence we proposed) is in general not a bound. On the other hand, it is often observed that the Bethe approximation is a more accurate approximation for the probability of evidence (see, e.g., Weiss, 2001).

Edge deletion in probabilistic graphical models.

Other methods of edge deletion have been proposed for Bayesian networks (Suermondt, 1992; van Engelen, 1997; Kjærulff, 1994), some of which can be re-phrased from a variational perspective. All of these approaches, however, approximate a network independent of the given evidence, which is a dramatic departure from ED-BP, and can lead to much worse behavior for less likely evidence. That is, these approaches approximate a network once for all queries, while ED-BP approximates a network for each specific query.

7 Conclusion

We have proposed a method for approximate inference in Bayesian networks, which reduces the problem into one of exact inference on an approximate network obtained by deleting edges. We have shown how the influential IBP algorithm corresponds to an instance of our framework, where one deletes enough edges from the network to render it a polytree, even a fully disconnected one. The proposed framework and its subsumption of IBP leads to a number of theoretical and practical implications that we explored in this paper.

On the theoretical side, our method leads to a new characterization of the fixed points of IBP, as well as the Bethe free energy approximation. This new perspective on IBP and its fundamental connection to the Bethe free energy is stated in terms of Bayesian networks alone (without appeal to concepts from statistical physics, for example). On the practical side, it leads to a practical and intuitive method for improving the quality of IBP approximations by

recovering those edges whose deletion is expected to worsen the quality of approximation the most. We hope that a simple edge deletion framework will make iterative belief propagation, and particularly its generalizations, more inviting to both practitioners and researchers in fields outside of artificial intelligence and information theory.

Acknowledgments

This work has been partially supported by Air Force grant #FA9550-05-1-0075 and by NSF grant #IIS-0713166.

APPENDIX

A Edge Deletion in Undirected Models

In this section, we highlight the semantics of edge deletion in pairwise Markov random fields (MRFs). The upcoming results, as well as their proofs, are analogous to the case for Bayesian networks, and can be extended to other undirected models, including factor graphs.

Let an MRF M have a graph $(\mathcal{E}, \mathcal{V})$ with edges $(i, j) \in \mathcal{E}$ and nodes $i \in \mathcal{V}$. Each node i of the graph is associated with a variable X_i taking on values x_i, with corresponding edge potentials $\psi(X_i, X_j)$ and node potentials $\psi(X_i)$. The distribution Pr induced by M is defined as follows:

$$Pr(\mathbf{x}) \stackrel{def}{=} \frac{1}{Z} \prod_{(i,j)\in\mathcal{E}} \psi(X_i{=}x_i, X_j{=}x_j) \prod_{i\in\mathcal{V}} \psi(X_i{=}x_i),$$

where $\mathbf{x}$ denotes an instantiation $x_1, \ldots, x_n$ of network variables and where Z is the *partition function:*

$$Z \stackrel{def}{=} \sum_{\mathbf{x}} \prod_{(i,j)\in\mathcal{E}} \psi(X_i{=}x_i, X_j{=}x_j) \prod_{i\in\mathcal{V}} \psi(X_i{=}x_i).$$

We will also be interested in computing marginal probabilities:

$$Pr(X_i{=}x_i) \stackrel{def}{=} \sum_{\mathbf{x}: X_i = x_i} Pr(\mathbf{x})$$

We shall assume that any observations on variables X_i are incorporated into node potentials $\psi(X_i)$.

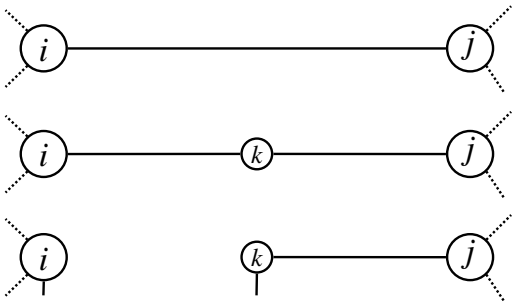


Fig. A.1. To delete an edge (top), we introduce an auxiliary node (middle), and delete the resulting equivalence edge, adding edge parameters (bottom).

An *equivalence edge* (i, j) in an MRF connects two variables X_i and X_j with the same domain, where the potential $\phi(X_i, X_j)$ denotes an equivalence constraint:

$$\phi(X_i{=}x_i, X_j{=}x_j) = \begin{cases} 1, \text{ if } x_i = x_j; \\ 0, \text{ otherwise.} \end{cases}$$

We now define the deletion of an equivalence edge, and show later that it is sufficient to assume that we only delete equivalence edges (equivalence edges were also used in Choi and Darwiche, 2006b).

Definition 2 (Edge Deletion) *Let (i, j) be an equivalence edge in a pairwise Markov random field M. We say that the equivalence edge (i, j) is deleted when it results in a network that is obtained from M as follows:*

- *The edge (i, j) is removed from the graph.*
- *Auxiliary potentials $\theta(X_i)$ and $\theta(X_j)$ are added for variables X_i and X_j.*

When deleting multiple edges, we may introduce multiple, yet distinct, potentials $\theta(X_i)$ for the same node X_i. We shall also collectively refer to auxiliary potentials $\theta(X_i)$ and $\theta(X_j)$ as *edge parameters*.

To delete an MRF edge (i, j) that is not an equivalence edge, we use the technique illustrated in Figure A.1:

- we introduce an auxiliary node k between i and j;
- introduce an equivalence constraint on the edge (i, k);
- copy the original potential of edge (i, j) to (k, j);
- and delete the equivalence edge (i, k).

Note that both the original model and the extended one will have the same treewidth in this case, and will also agree on the distribution over their common variables.

Now, let an MRF M have edges $\mathcal{E} \cup \mathcal{E}'$, where $\mathcal{E}'$ is a set of equivalence edges

that we wish to delete. We can now define the distribution Pr induced by M, and the distribution Pr' induced by the resulting approximation M', as follows:

$$Pr(\mathbf{x}) = \frac{1}{Z}\Psi(\mathbf{x})\Phi(\mathbf{x})$$
$$Pr'(\mathbf{x};\Theta) = \frac{1}{Z'(\Theta)}\Psi(\mathbf{x})\Theta(\mathbf{x})$$

where

- $\Psi(\mathbf{x}) = \prod_{(i,j)\in\mathcal{E}} \psi(X_i{=}x_i, X_j{=}x_j) \prod_{i\in\mathcal{V}} \psi(X_i{=}x_i)$ are the original network potentials;
- $\Phi(\mathbf{x}) = \prod_{(i,j)\in\mathcal{E}'} \phi(X_i{=}x_i, X_j{=}x_j)$ are the equivalence edges deleted in M;
- $\Theta(\mathbf{x}) = \prod_{(i,j)\in\mathcal{E}'} \theta(X_i{=}x_i)\theta(X_j{=}x_j)$ are the edge parameters added to M';

Moreover, the partition function Z and the approximate partition function Z' are now defined as:

$$Z = \sum_{\mathbf{x}} \Psi(\mathbf{x})\Phi(\mathbf{x}) \qquad \text{and} \qquad Z'(\Theta) = \sum_{\mathbf{x}} \Psi(\mathbf{x})\Theta(\mathbf{x}).$$

When we make a specific choice of parameters Θ, we shall simply refer to the distribution $Pr'(.)$, leaving the choice of Θ implicit.

In an undirected model, we can think of deleting an equivalence edge (i, j) as ignoring a constraint that equates variables X_i and X_j. The edge parameters may then be used to compensate for this lost equivalence, by forcing some weaker notion of equivalence between these variables.

Since we are deleting an equivalence constraint X_i and X_j, remember that they take on the same states x. Consider then the following theorem, which is analogous to Theorems 1, 2 and 3 for Bayesian networks.

Theorem 5 *Let M be a pairwise MRF, and let M' be the result of deleting equivalence edges $(i, j) \in \mathcal{E}'$ in M. The following two conditions are equivalent:*

(C1) *Equivalence between marginals for variables X_i and X_j, which are computable from edge parameters: for all states x of X_i and X_j:*

$$Pr'(X_i{=}x) = Pr'(X_j{=}x) \propto \theta(X_i{=}x)\theta(X_j{=}x).$$

(C2) *Fixed point conditions on edge parameters:*

$$\theta(X_i{=}x) = \alpha\frac{\partial Z'}{\partial\theta(X_j{=}x)} \qquad \theta(X_j{=}x) = \alpha\frac{\partial Z'}{\partial\theta(X_i{=}x)},$$

for all states x, where α is a normalizing constant.

Moreover, if M' has a tree structure, then for every IBP fixed point in M, there exists a set of edge parameters that satisfy the above conditions for M' (and vice versa) where:

- $b(X_i{=}x_i) = Pr'(X_i{=}x_i)$, *for each* $i \in \mathcal{V}$;
- $b(X_i{=}x_i, X_j{=}x_j) = Pr'(X_i{=}x_i, X_j{=}x_j)$, *for each* $(i,j) \in \mathcal{E}$;

where $b(.)$ *are the approximate marginals given by IBP.*

We assume here that $\theta(X_i)$ and $\theta(X_j)$ each sum to one. Note again that the partial derivatives of Condition (C2) can be computed efficiently in traditional inference frameworks (Darwiche, 2003; Park and Darwiche, 2004). Condition (C2) further suggests an ED-BP algorithm for parametrizing edges in MRFs, analogous to Algorithm 1 for Bayesian networks.

In a pairwise MRF, the Bethe free energy F_β is defined as follows:

$$F_\beta = U_\beta - H_\beta$$

where U_β is the Bethe *average* energy

$$U_\beta = -\sum_{(i,j)\in\mathcal{E}} E[\,\log \psi(X_i, X_j)\,] - \sum_{i\in\mathcal{V}} E[\,\log \psi(X_i)\,]$$

and H_β is the Bethe *approximate* entropy

$$H_\beta = -\sum_{(i,j)\in\mathcal{E}} H(X_i, X_j) + \sum_{i\in\mathcal{V}} (n_i - 1) H(X_i),$$

where n_i is the number of neighbors of node i, and where the expectations $E[\,.\,]$ and entropies $H(.)$ are with respect to the marginal approximations $b(X_i, X_j)$ and $b(X_i)$ given by IBP. Under certain consistency and normalization constraints, one can show that the approximate marginals are a stationary point of the Bethe free energy if and only if IBP messages are a fixed point of iterative belief propagation (Yedidia et al., 2005). We can further use F_β as an approximation to the partition function, i.e., $Z_\beta = \exp\{-F_\beta\}$. As in Theorem 6, this approximation is an edge-corrected partition function.

Theorem 6 *Let M be a pairwise MRF, and let M' be the result of deleting a set of equivalence edges $\mathcal{E}'$ in M with* ED-BP *edge parameters, but further that M' is rendered a tree. Then*

$$Z_\beta = Z' \cdot \frac{1}{z} \qquad \textit{where} \qquad z = \prod_{(i,j)\in\mathcal{E}'} z_{ij} = \prod_{(i,j)\in\mathcal{E}'} \sum_x \theta(X_i{=}x)\theta(X_j{=}x).$$

When M' is not a tree, the edge-corrected partition function $Z' \cdot \frac{1}{z}$ corresponds to an approximation $Z_{JG} = \exp\{-F_{JG}\}$ of a particular joingraph free energy F_{JG} (Yedidia et al., 2005; Aji and McEliece, 2001); see Appendix D.

As in Proposition 2, we can instead characterize the fixed points of ED-BP (and hence the fixed points of IBP) as a stationary point of an edge-corrected partition function.

Proposition 3 *Let M be a pairwise MRF, and let M' be the result of deleting a set of equivalence edges $\mathcal{E}'$ in M with* ED-BP *edge parameters. Edge parameters of M' are an* ED-BP *fixed point if and only if the edge parameters are a stationary point of $Z' \cdot \frac{1}{z}$.*

B Proofs for Section 3

In the main text, we opted to refer to edge parameters by $PM(\hat{U})$ and $SE(U)$. In our proofs, in contrast, we shall typically refer directly to the corresponding CPTs $\Theta_{\hat{U}}$ and $\Theta_{\hat{S}|U}$. More specifically, we shall usually refer to the CPT parameter $\theta_{\hat{u}}$ for the edge parameter $PM(\hat{U}=\hat{u})$, and the CPT parameter $\theta_{\hat{s}|u}$ for the edge parameter $SE(U=u)$. Moreover, when we set variables $\hat{U}$ and U to the same state u, we will refer to the CPT parameters θ_u and $\theta_{\hat{s}|u}$ instead of the edge parameters $PM(\hat{U}=u)$ and $SE(U=u)$. This choice of notation should simplify the presentation of our proofs, and should be unambiguous in context.

We note now that the conditions on the ED-BP edge parameters in Theorem 1 have the following probabilistic semantics:

$$\theta_u = \alpha \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_{\hat{s}|u}} = \alpha \frac{Pr'(U=u, \mathbf{e}')}{\theta_{\hat{s}|u}} = Pr'(U=u \mid \mathbf{e}' \setminus \hat{s}) \tag{B.1}$$

$$\theta_{\hat{s}|u} = \alpha \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_u} = \alpha \frac{Pr'(\hat{U}=u, \mathbf{e}')}{\theta_u} = \alpha Pr'(\mathbf{e}' \mid \hat{U}=u) \tag{B.2}$$

Again, θ_u is the CPT parameter for variable $\hat{U}$ corresponding to the edge parameter $PM(\hat{U}=u)$, and $\theta_{\hat{s}|u}$ is the CPT parameter for variable $\hat{S}$ corresponding to the edge parameter $SE(U=u)$.

Throughout our proofs, when it is convenient, we shall interchange among the equivalent characterizations of ED-BP given by Condition 1, Theorem 1, and Equations B.1 and B.2 above.

Proof of Theorem 1 We first show Condition 1 implies the fixed point conditions given by Equations 3 and 4. Note first that Equation B.1 tells us that $\theta_u = Pr'(\hat{U}=u) = Pr'(U=u|\mathbf{e}' \setminus \hat{s})$, so Equations 2 and 3 are equivalent. Thus it suffices to show, given Equations 2 and 3, that Equation 1 holds if and only

if Equation 4 holds:

$$
\begin{aligned}
& \theta_{\hat{s}|u} = \alpha \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_u} && \text{by Eqs. 4, B.2} \\
\iff & \theta_{\hat{s}|u} \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_{\hat{s}|u}} = \alpha \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_{\hat{s}|u}} \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_u} \\
\iff & \theta_{\hat{s}|u} \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_{\hat{s}|u}} = \theta_{\hat{u}} \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_u} && \text{by Eqs. 3, B.1} \\
\iff & Pr'(U{=}u, \mathbf{e}') = Pr'(\hat{U}{=}u, \mathbf{e}') \\
\iff & Pr'(U{=}u \mid \mathbf{e}') = Pr'(\hat{U}{=}u \mid \mathbf{e}').
\end{aligned}
$$

Note that the normalizing constant α for θ_u and $\theta_{\hat{s}|u}$ are in fact the same value, $\alpha = [Pr'(\mathbf{e}' \setminus \hat{s})]^{-1}$ (see Eq. B.1), for a network satisfying Condition 1. To see why, let α_{SE} and α_{PM} denote the normalizing constants for $\theta_{\hat{s}|u}$ and θ_u respectively. First, we multiply Equations 3 and 4 together:

$$\alpha_{SE} \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_u} \cdot \theta_u = \alpha_{PM} \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_{\hat{s}|u}} \cdot \theta_{\hat{s}|u}$$

and thus $\alpha_{SE} \cdot Pr'(\hat{U}{=}u, \mathbf{e}') = \alpha_{PM} \cdot Pr'(U{=}u, \mathbf{e}')$. Summing for all states u, we have $\alpha_{SE} Pr'(\mathbf{e}') = \alpha_{PM} Pr'(\mathbf{e}')$, and thus $\alpha_{SE} = \alpha_{PM} = \alpha$. □

Proof of Lemma 1 By Theorem 1, Condition 1 is equivalent to an ED-BP fixed point given by Equations 3 and 4. Thus, we show an ED-BP fixed point is necessary and sufficient for exact marginals when deleting an edge splits a network into two independent subnetworks.

First, let $\hat{S}{=}\hat{s}$ be the soft evidence introduced when deleting the edge $U \to X$. Let $\mathbf{e}_U, \hat{s}$ denote the evidence instantiated in the subnetwork of N' containing variable U, and let $\mathbf{e}_X$ denote the evidence in the subnetwork containing variable X. Now, in the original network N we have:

$$
\begin{aligned}
& Pr(\mathbf{x}_U, \mathbf{e}_U) Pr(\mathbf{e}_X | U{=}u) \\
& = Pr(\mathbf{x}_U, \mathbf{e}_U | U{=}u) Pr(\mathbf{e}_X | U{=}u) Pr(U{=}u) \\
& = Pr(\mathbf{x}_U, \mathbf{e} | U{=}u) Pr(U{=}u) && \text{by d–separation} \\
& \propto Pr(\mathbf{x}_U | \mathbf{e})
\end{aligned}
$$

In the approximate network N' we have:

$$
\begin{aligned}
Pr'(\mathbf{x}_U, \mathbf{e}_U) \theta_{\hat{s}|u} & \propto Pr'(\mathbf{x}_U | \mathbf{e}_U, \hat{s}) \\
& = Pr'(\mathbf{x}_U | \mathbf{e}') && \text{by d–separation}
\end{aligned}
$$

Observe that after pruning nodes and edges in N and N' for queries $Pr(\mathbf{x}_U, \mathbf{e}_U)$ and $Pr'(\mathbf{x}_U, \mathbf{e}_U)$, the resulting networks are equivalent. Thus, we know that $Pr(\mathbf{x}_U, \mathbf{e}_U) = Pr'(\mathbf{x}_U, \mathbf{e}_U)$. Similarly, $Pr(\mathbf{e}_X | U{=}u) = Pr'(\mathbf{e}_X | \hat{U}{=}u)$. In an

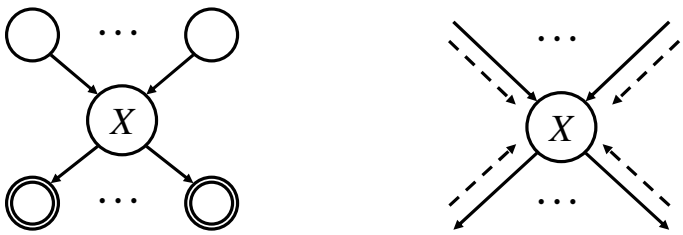


Fig. B.1. Correspondence between parameter updates ED-BP in a fully disconnected network and message passing in IBP.

ED-BP network, $\theta_{\hat{s}|u} \propto Pr'(\mathbf{e}_X|\hat{U}=u)$ (Eq. B.2), so we can conclude that $Pr(\mathbf{x}_U|\mathbf{e}) = Pr'(\mathbf{x}_U|\mathbf{e}')$.

Similarly to show that $Pr(\mathbf{x}_X|\mathbf{e}') = Pr'(\mathbf{x}_X|\mathbf{e}')$ holds if and only if Equation B.1 holds. □

Proof of Theorem 2 We shall show the correspondence between ED-BP and IBP, by induction. Let a variable X in network N have parents U_i and children Y_j. Observe that X is the center of a star in N' whose arms are auxiliary variables $\hat{U}_i$ and $\hat{S}_j$ introduced by deleting edges $U_i \rightarrow X$ and $X \rightarrow Y_j$, respectively; see Figure B.1 (left). For an iteration t, let $PM_i^t(\hat{U}_i)$ parametrize clone variable $\hat{U}_i$ and let $SE_j^t(X)$ parametrize soft evidence variable $\hat{S}_j$. Then at iteration $t = 0$, we are given that $\pi_X^0(U_i) = PM_i^0(\hat{U}_i)$ for all edges $U_i \rightarrow X$, and $\lambda_{Y_j}^0(X) = SE_j^0(X)$ for all edges $X \rightarrow Y_j$.

We first want to show, for an iteration $t > 0$ and for an edge $X \rightarrow Y_j$, that the IBP message that variable X passes to its child Y_j is the same as the parameters for the clone $\hat{X}$ that was made a parent of Y_j. That is, we want to show $\pi_{Y_j}^t(X) = PM_j^t(\hat{X})$. Assume for notation that all evidence $\mathbf{e}$ is virtual in N, and that the evidence in the star of X in N' is $\mathbf{e}'_X$. Starting from Equation B.1, we have:

$$PM_j^t(\hat{X}=x) = Pr'_{t-1}(X=x \mid \mathbf{e}' \setminus \hat{s}_j) = Pr'_{t-1}(X=x \mid \mathbf{e}'_X \setminus \hat{s}_j),$$

since X is independent of all evidence other than the evidence $\mathbf{e}'_X$ that is directly connected to X. Letting $\hat{\mathbf{u}}$ denote an instantiation of the clones that became parents of X in N', we have

$$PM_j^t(\hat{X}=x) = \alpha Pr'_{t-1}(X=x, \mathbf{e}'_X \setminus \hat{s}_j) = \alpha \sum_{\hat{\mathbf{u}}} Pr'_{t-1}(X=x, \hat{\mathbf{u}}, \mathbf{e}'_X \setminus \hat{s}_j).$$

We then factorize into the subnetwork parameters of the star centered at X:

$$\begin{aligned} PM_j^t(\hat{X}=x) &= \alpha \sum_{\hat{\mathbf{u}}} \theta_{x|\hat{\mathbf{u}}} \prod_i \theta_{\hat{u}_i}^{t-1} \prod_{k \neq j} \theta_{\hat{s}_k|x}^{t-1} \\ &= \alpha \sum_{\hat{\mathbf{u}}} \theta_{x|\hat{\mathbf{u}}} \prod_i PM_i^{t-1}(\hat{U}_i=\hat{u}_i) \prod_{k \neq j} SE_k^{t-1}(X=x). \end{aligned}$$

Finally, by our inductive hypothesis, we have the desired BP update equation:

$$PM_j^t(\hat{X}=x) = \alpha \sum_{\mathbf{u}} \theta_{x|\mathbf{u}} \prod_i \pi_X^{t-1}(U_i=u_i) \prod_{k \neq j} \lambda_{Y_j}^{t-1}(X=x) = \pi_{Y_j}^t(X=x).$$

Similarly to show that $\lambda_{Y_j}^t(X) = SE_j^t(X)$, and that $b_t(X|\mathbf{e}) = Pr'_t(X|\mathbf{e}')$. □

Proof of Theorem 3 It suffices to show that an ED-BP fixed point and its node marginals are invariant to the deletion or recovery of a single edge, as long as the approximation is a polytree before and after the operation. Any polytree network N' can then be transformed to another polytree network N'' via a finite sequence of deletion and recovery operations.

A single edge deleted. First, observe that since N' is a polytree, deleting any edge splits the network into two independent subnetworks. If we fix the parameters for edges already deleted, then the resulting network, N'', will have the same node marginals as N' by Lemma 1. Moreover, all edges originally deleted in N' will continue to satisfy Condition 1, and thus N'' will also be an ED-BP approximation with respect to the original network N.

To see why, consider an edge $U \rightarrow X$ that was deleted in N'. Clearly, Equation 1 of Condition 1 continues to hold in N''. As for Equation 2 consider the fact that

$$\begin{aligned} &Pr'(U=u \mid \mathbf{e}') \propto Pr'(U=u, \mathbf{e}') \\ &= Pr'(U=u, \mathbf{e}' \setminus \hat{s}) \cdot \theta_{\hat{s}|u} \propto Pr'(U=u \mid \mathbf{e}' \setminus \hat{s}) \cdot \theta_{\hat{s}|u}. \end{aligned}$$

Since the value of $Pr'(U=u|\mathbf{e}')$ does not change going from network N' to network N'', and since we left $\theta_{\hat{s}|u}$ fixed, the value of $Pr'(U=u|\mathbf{e}' \setminus \hat{s})$ does not change. Further, since the values of $Pr'(U=u|\mathbf{e}' \setminus \hat{s})$ and $Pr'(\hat{U}=\hat{u}) = \theta_{\hat{u}}$ do not change, this edge must satisfy Condition 2 in network N'' as it did in N'.

A single edge recovered. Say we recover an edge $U \rightarrow X$ into N', leaving all edge parameters fixed, giving us network N''. We can then observe that N' is simply the result of deleting the single edge $U \rightarrow X$ from N''. Arguing as before, it must be that N'' must have the same marginals as N', and further that edges deleted in N'' continue to satisfy Condition 1. □

Theorems 2 and 3 state a correspondence between IBP node marginals and ED-BP node marginals in polytree approximations. It is also straightforward to show that approximations to family marginals are also the same. In particular, if X has parents $\mathbf{U}$ in network N, and X has parents $\mathbf{U}'$ in N' (some of which are cloned parents), the IBP marginals for a family $X\mathbf{U}$ in N are the ED-BP marginals for a family $X\mathbf{U}'$ in N'.

Corollary 2 *Let N be a Bayesian network and let N' be a polytree found by deleting edges from N. Suppose we have an IBP fixed point in N and a*

corresponding ED-BP *fixed point in* N'*. Then, for each family* $X\mathbf{U}$ *in* N *and the corresponding family* $X\mathbf{U}'$ *in* N'*, we have*

- $b(X\mathbf{U} \mid \mathbf{e}) = Pr'(X\mathbf{U}' \mid \mathbf{e}')$.

C Proofs for Section 4

First, we derive two identities that will be useful to prove our theorems. Note that we assume here that the original network N is strictly positive.

Lemma 2 *Let* N *be a Bayesian network and* N' *be the result of deleting a single edge* $U \rightarrow X$ *from* N*. Then:*

$$Pr(\mathbf{e}) = \sum_{u} \frac{\partial^2 Pr'(\mathbf{e}')}{\partial \theta_u \partial \theta_{\hat{s}|u}} \tag{C.1}$$

Proof Say, that we replace the edge $U \rightarrow X$ in N with a chain $U \rightarrow \hat{U} \rightarrow X$ where the edge $U \rightarrow \hat{U}$ denotes an equivalence constraint: $\theta_{\hat{u}|u} = 1$ iff $\hat{u} = u$. The resulting augmented network is equivalent to the original network over the original variables, and in particular, it would yield the same probability of evidence. Rather than deleting edge $U \rightarrow X$, we can instead delete the equivalence edge $U \rightarrow \hat{U}$, adding only the auxiliary variable $\hat{S}$ as the clone variable $\hat{U}$ is already present. We thus observe:

$$\begin{aligned} Pr(\mathbf{e}) &= \sum_{u\hat{u}} Pr(U{=}u, \hat{U}{=}\hat{u}, \mathbf{e}) \\ &= \sum_{u\hat{u}} \frac{\partial Pr(\mathbf{e})}{\partial \theta_{\hat{u}|u}} \theta_{\hat{u}|u} = \sum_{u=\hat{u}} \frac{\partial Pr(\mathbf{e})}{\partial \theta_{\hat{u}|u}} = \sum_{u=\hat{u}} \frac{\partial^2 Pr'(\mathbf{e}')}{\partial \theta_{\hat{u}} \partial \theta_{\hat{s}|u}} = \sum_{u} \frac{\partial^2 Pr'(\mathbf{e}')}{\partial \theta_u \partial \theta_{\hat{s}|u}} \end{aligned}$$

as desired. □

Lemma 3 *Let* N *be a Bayesian network and* N' *be the result of deleting a single edge* $U \rightarrow X$ *from* N*. Then* $MI(U; \hat{U}|\mathbf{e}') = 0$ *in* N' *if and only if*

$$\frac{\partial^2 Pr'(\mathbf{e}')}{\partial \theta_u \partial \theta_{\hat{s}|u}} = \frac{1}{Pr'(\mathbf{e}')} \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_u} \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_{\hat{s}|u}} \tag{C.2}$$

Proof ($\Leftarrow$) First, by multiplying $[Pr'(\mathbf{e}')]^{-1} \cdot \theta_u \theta_{\hat{s}|u}$ on both sides of Equation C.2, we find that:

$$\frac{1}{Pr'(\mathbf{e}')} \frac{\partial^2 Pr'(\mathbf{e}')}{\partial \theta_u \partial \theta_{\hat{s}|u}} \theta_u \theta_{\hat{s}|u} = \frac{1}{Pr'(\mathbf{e}')} \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_u} \theta_u \cdot \frac{1}{Pr'(\mathbf{e}')} \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_{\hat{s}|u}} \theta_{\hat{s}|u}, \tag{C.3}$$

which simplifies to

$$Pr'(U{=}u, \hat{U}{=}u \mid \mathbf{e}') = Pr'(U{=}u \mid \mathbf{e}') Pr'(\hat{U}{=}u \mid \mathbf{e}'). \tag{C.4}$$

Thus, U is independent of $\hat{U}$ given $\mathbf{e}'$, and $MI(U;\hat{U}|\mathbf{e}') = 0$.

($\Rightarrow$) If $MI(U;\hat{U}|\mathbf{e}') = 0$, then we have Equations C.4 and Equation C.3. If all edge parameters are non-negative, then Equation C.2 follows trivially. □

Proof of Proposition 1 First, employing Equation 3, we have

$$z_{UX} = \sum_u \theta_u \theta_{\hat{s}|u} = \sum_u \alpha \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_{\hat{s}|u}} \theta_{\hat{s}|u} = \alpha \sum_u Pr'(U{=}u, \mathbf{e}') = \alpha Pr'(\mathbf{e}'). \quad \text{(C.5)}$$

It remains to show that $\alpha = [Pr(\mathbf{e})]^{-1}$. Using Lemmas 2 and 3, we have:

$$Pr(\mathbf{e}) = \sum_u \frac{\partial^2 Pr'(\mathbf{e}')}{\partial \theta_u \partial \theta_{\hat{s}|u}} = \sum_u \frac{1}{Pr'(\mathbf{e}')} \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_u} \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_{\hat{s}|u}}.$$

Substituting in $\frac{\partial Pr'(\mathbf{e}')}{\partial \theta_u} = \frac{\theta_{\hat{s}|u}}{\alpha}$ from Equation 3, we have

$$Pr(\mathbf{e}) = \frac{1}{\alpha} \sum_u \frac{\theta_{\hat{s}|u}}{Pr'(\mathbf{e}')} \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_{\hat{s}|u}} = \frac{1}{\alpha} \sum_u Pr'(U{=}u \mid \mathbf{e}') = \frac{1}{\alpha}$$

as desired. Note, from the proof of Theorem 1, that the normalizing constant α is in fact the same value for both $\theta_{\hat{u}}$ and $\theta_{\hat{s}|u}$. In particular, $\alpha = [Pr'(\mathbf{e}' \setminus \hat{s})]^{-1}$, which is simply $Pr'(\mathbf{e})$ when a single edge is deleted. □

Proof of Theorem 4 When a given network is a polytree, the Bethe free energy is the exact free energy, and yields the exact probability of evidence. Consider then the Bethe free energy F_β of the original network N and the exact free energy F' of the approximate network N'. Since $Pr_\beta(\mathbf{e}) = \exp\{-F_\beta\}$ and $Pr'(\mathbf{e}') = \exp\{-F'\}$, in order to show that $Pr_\beta(\mathbf{e}) = Pr'(\mathbf{e}') \cdot \frac{1}{z}$, we need to show that $-F_\beta = -F' - \log z$, or equivalently $F' = F_\beta - \log z$.

Let $E[\,.\,]$ denote expectations and $H'(.)$ denote entropies, both with respect to $Pr'(.)/b(.)$, which we use interchangeably since Theorem 2 and Corollary 2 tell us that family and node marginals of N' are IBP beliefs.

Since N' is fully-disconnected, the components of its exact free energy $F' = U' - H'$ are simply:

$$U' = -\sum_{X\hat{\mathbf{U}}} E[\,\log \Theta_{X|\hat{\mathbf{U}}} \lambda_X\,] - \sum_{U \to X} E[\,\log \Theta_{\hat{U}} \Theta_{\hat{s}|U}\,]$$
$$H' = \sum_{X\hat{\mathbf{U}}} H'(X\hat{\mathbf{U}} \mid \mathbf{e}').$$

Here, $X\hat{\mathbf{U}}$ denotes a family in N' (a variable X and its cloned parents $\hat{\mathbf{U}}$). Note that when we apply Equation 11 to network N', the entropies $H'(\hat{U} \mid \mathbf{e}')$ in the family term cancel out the same entropies appearing in the parent term.

Now, since

$$\theta_u \theta_{\hat{s}|u} = \alpha \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_{\hat{s}|u}} \theta_{\hat{s}|u} = \alpha Pr'(\mathbf{e}') Pr'(U{=}u \mid \mathbf{e}') = z_{UX} Pr'(U{=}u \mid \mathbf{e}')$$

(see Equations 3 and C.5), we have

$$\begin{aligned}
\sum_{U \to X} E[\log \Theta_{\hat{U}} \Theta_{\hat{S}|U}] &= \sum_{U \to X} \Big[\log z_{UX} - H'(U \mid \mathbf{e}') \Big] \\
&= \sum_{U \to X} \log z_{UX} - \sum_{U \to X} H'(U \mid \mathbf{e}') \\
&= \log z - \sum_{X\mathbf{U}} \sum_{U \in \mathbf{U}} H'(U \mid \mathbf{e}').
\end{aligned}$$

After substituting into U', we have the exact free energy for network N':

$$\begin{aligned}
&F' = U' - H' \\
&= \Big[- \sum_{X\hat{\mathbf{U}}} E[\log \Theta_{X|\hat{\mathbf{U}}} \lambda_X] - \log z + \sum_{X\mathbf{U}} \sum_{U \in \mathbf{U}} H'(U \mid \mathbf{e}') \Big] - \Big[\sum_{X\hat{\mathbf{U}}} H'(X\hat{\mathbf{U}} \mid \mathbf{e}') \Big]
\end{aligned}$$

We then collect the terms that make up the Bethe free energy $F_\beta = U_\beta - H_\beta$:

$$\begin{aligned}
&F' = U' - H' \\
&= \Big[- \sum_{X\hat{\mathbf{U}}} E[\log \Theta_{X|\hat{\mathbf{U}}} \lambda_X] \Big] - \Big[\sum_{X\hat{\mathbf{U}}} H'(X\hat{\mathbf{U}} \mid \mathbf{e}') - \sum_{X\mathbf{U}} \sum_{U \in \mathbf{U}} H'(U \mid \mathbf{e}') \Big] - \log z \\
&= U_\beta - H_\beta - \log z = F_\beta - \log z
\end{aligned}$$

giving us the desired result.

We now show that the property $F' = F_\beta - \log z$ persists when we recover edges into N', as long as the resulting network remains a polytree. When we recover a single edge $U \to X$ into N', the resulting network N'' has one less pair of edge parameters, so the resulting average energy U'', in terms of the average energy U' for N', is:

$$U'' = U' + E[\log \Theta_{\hat{U}} \Theta_{\hat{s}|U}] = U' + \log z_{UX} - H'(U \mid \mathbf{e}').$$

Since network N'' has an additional edge $U \to X$, the resulting entropy H'' loses the family entropy and the parent entropy terms for the clone $\hat{U}$ (which canceled each other out), and gains a new parent entropy for the original parent U. The resulting entropy, in terms of the previous entropy H', is then $H'' = H' - H'(U|\mathbf{e}')$. Noting that the newly introduced entropies $H'(U|\mathbf{e}')$ in the average energy and the approximate entropy cancel each other out, the new free energy in the network N'' where an edge has been recovered into N'

is

$$\begin{aligned} F'' &= U'' - H'' \\ &= U' - H' + \log z_{UX} \\ &= U_\beta - H_\beta - \log z + \log z_{UX} \\ &= F_\beta - [\log z - \log z_{UX}] \end{aligned}$$

where $\frac{z}{z_{UX}}$ is the new correction factor for network N'' (we no longer have to correct for the edge $U \rightarrow X$). Since node and family marginals do not change in N' and N'', the corrected probability of evidence is still equivalent to the Bethe approximation for N, as long as the resulting network remains a polytree. □

Proof of Proposition 2 For simplicity, we implicitly assume that edge parameters will sum to one, i.e., $\sum_{\hat{u}} \theta_{\hat{u}} = \sum_u \theta_{\hat{s}|u} = 1$. One could otherwise explicitly constructing the Lagrangian, with appropriate adjustments to the proof below.

First, we show that a stationary point of $f = Pr'(\mathbf{e}') \cdot \frac{1}{z}$ is an ED-BP fixed point, as in Theorem 1. Consider a pair of edge parameters $\theta_u, \theta_{\hat{s}|u}$. Setting to zero the partial derivatives of f with respect to a parameter θ_u, we have:

$$\frac{\partial f}{\partial \theta_u} = \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_u} \cdot \frac{1}{z} - \frac{Pr'(\mathbf{e}')}{z^2} \cdot \frac{\partial z}{\partial \theta_u} = 0. \tag{C.6}$$

Letting $z_{UX} = \sum_u \theta_u \theta_{\hat{s}|u}$ we further have

$$\frac{\partial z}{\partial \theta_u} = \frac{z}{z_{UX}} \frac{\partial z_{UX}}{\partial \theta_u} = \frac{z}{z_{UX}} \theta_{\hat{s}|u}.$$

Substituting into Equation C.6, and then multiplying by z:

$$\frac{\partial f}{\partial \theta_u} = \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_u} - \frac{Pr'(\mathbf{e}')}{z_{UX}} \cdot \theta_{\hat{s}|u} = 0.$$

Rearranging, we find that:

$$\theta_{\hat{s}|u} = \frac{z_{UX}}{Pr'(\mathbf{e}')} \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_u}. \tag{C.7}$$

Summing these equations for all states u , and employing our normalization constraint $\sum_u \theta_{\hat{s}|u} = 1$, we have

$$1 = \frac{z_{UX}}{Pr'(\mathbf{e}')} \sum_u \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_u}$$

or equivalently

$$\frac{z_{UX}}{Pr'(\mathbf{e}')} = \left[\sum_u \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_u} \right]^{-1} = \alpha$$

Substituting into Equation C.7, we have

$$\theta_{\hat{s}|u} = \alpha \frac{\partial Pr'(\mathbf{e}')}{\partial \theta_u}$$

as desired. We can derive the ED-BP update equation for θ_u in Equation 3 in the same way, thus showing a stationary point of a corrected probability of evidence is a fixed point of ED-BP. Plugging in our update equations in Equation C.6, we can similarly show that an ED-BP fixed point is a zero gradient of f. □

D ED-BP and IJGP

ED-BP can in fact be simulated by a particular class of GBP algorithms called iterative joingraph propagation (IJGP) (Aji and McEliece, 2001; Dechter et al., 2002). To specify an IJGP approximation, one must specify a joingraph on which message passing is performed, which can by itself be a non-trivial task. ED-BP can then be considered a means of designing an IJGP approximation, where deleting edges in the model implies a particular choice of a joingraph. We sketch the correspondence between ED-BP and IJGP below.

Let N be a Bayesian network and N' be the result of deleting edges $U \rightarrow X$ from N. To parametrize N' using ED-BP, we can use a jointree algorithm for exact inference in N'. Further, we can choose a jointree where the CPTs $\Theta_{\hat{U}}$ and $\Theta_{\hat{S}|U}$ are each assigned to unique leaf clusters (we can easily construct one from any jointree for N', by introducing auxiliary clusters). Finally, for each edge $U \rightarrow X$ deleted, we can connect in the jointree the two leaf clusters containing $\Theta_{\hat{U}}$ and $\Theta_{\hat{S}|U}$. When we relabel the clones $\hat{U}$ by the label U, we have a joingraph that we can use for IJGP.

We can then simulate ED-BP by running IJGP in our specially constructed joingraph. We initialize IJGP messages in the same way we initialized edge parameters in ED-BP. Each IJGP iteration t then consists of two message-passing phases. First, we propagate messages (pull-push propagation) in the original jointree embedded in our joingraph; this corresponds to propagating messages in a jointree for N'_{t-1}. Second, we propagate messages in parallel across the remaining edges that connect the clusters containing $\Theta_{\hat{U}}$ and $\Theta_{\hat{S}|U}$; this corresponds to computing parameters $PM_t(\hat{U})$ and $SE_t(U)$ for network N'_t by performing inference in network N'_{t-1} (see Equations 5 and 6). Note that a jointree message incoming a leaf cluster, which is assigned a unique CPT, is equivalent to the partial derivative with respect to that CPT (as those that are required to compute edge parameters).

In Section 4, we saw that an edge-corrected probability of evidence corre-

sponds to the Bethe free energy approximation when the simplified network is a polytree. In general, an edge-corrected probability of evidence corresponds to a joingraph free energy (Aji and McEliece, 2001), induced from a joingraph constructed in the way we described above.

References

Aji, S. M., McEliece, R. J., 2001. The generalized distributive law and free energy minimization. In: Proceedings of the 39th Allerton Conference on Communication, Control and Computing. pp. 672–681.

Braunstein, A., Mézard, M., Zecchina, R., 2005. Survey propagation: An algorithm for satisfiability. Random Struct. Algorithms 27 (2), 201–226.

Chan, H., Darwiche, A., 2005. On the revision of probabilistic beliefs using uncertain evidence. Artificial Intelligence 163, 67–90.

Chavira, M., Darwiche, A., 2006. Encoding CNFs to empower component analysis. In: Proceedings of the 9th International Conference on Theory and Applications of Satisfiability Testing (SAT). Springer Berlin / Heidelberg, Lecture Notes in Computer Science, Volume 4121, pp. 61–74.

Choi, A., Chan, H., Darwiche, A., 2005. On Bayesian network approximation by edge deletion. In: Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence (UAI). AUAI Press, Arlington, Virginia, pp. 128–135.

Choi, A., Darwiche, A., 2006a. An edge deletion semantics for belief propagation and its practical impact on approximation quality. In: Proc. AAAI National Conference. pp. 1107–1114.

Choi, A., Darwiche, A., 2006b. A variational approach for approximating Bayesian networks by edge deletion. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence. pp. 80–89.

Choi, A., Darwiche, A., 2008a. Approximating the partition function by deleting and then correcting for model edges. In: Proceedings of the 24th Conference on Uncertainty in Artificial Intelligence (UAI). pp. 79–87.

Choi, A., Darwiche, A., 2008b. Focusing generalizations of belief propagation on targeted queries. In: Proceedings of the 23rd AAAI Conference on Artificial Intelligence (AAAI). pp. 1024–1030.

Choi, A., Darwiche, A., 2008c. Many-pairs mutual information for adding structure to belief propagation approximations. In: Proceedings of the 23rd AAAI Conference on Artificial Intelligence (AAAI). pp. 1031–1036.

Cover, T. M., Thomas, J. A., 1991. Elements of information theory. Wiley-Interscience.

Darwiche, A., 2001. Recursive conditioning. Artificial Intelligence 126 (1-2), 5–41.

Darwiche, A., 2003. A differential approach to inference in bayesian networks. Journal of the ACM 50 (3), 280–305.

Darwiche, A., Dechter, R., Choi, A., Gogate, V., Otten, L., 2008.

Results from the probablistic inference evaluation of UAI-08, http://graphmod.ics.uci.edu/uai08/Evaluation/Report.

Dechter, R., 1996. Bucket elimination: A unifying framework for probabilistic inference. In: Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence (UAI). pp. 211–219.

Dechter, R., Kask, K., Mateescu, R., 2002. Iterative join-graph propagation. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence. pp. 128–136.

Elidan, G., McGraw, I., Koller, D., 2006. Residual belief propagation: Informed scheduling for asynchronous message passing. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence. pp. 165–173.

Felzenszwalb, P. F., Huttenlocher, D. P., 2006. Efficient belief propagation for early vision. International Journal of Computer Vision 70 (1), 41–54.

Frey, B. J., MacKay, D. J. C., 1997. A revolution: Belief propagation in graphs with cycles. In: Advances in Neural Information Processing Systems (NIPS). pp. 479–485.

Frey, B. J., Patrascu, R., Jaakkola, T., Moran, J., 2000. Sequentially fitting "inclusive" trees for inference in noisy-or networks. In: Advances in Neural Information Processing Systems (NIPS). pp. 493–499.

Geiger, D., Meek, C., Wexler, Y., 2006. A variational inference procedure allowing internal structure for overlapping clusters and deterministic constraints. J. Artif. Intell. Res. (JAIR) 27, 1–23.

Ghahramani, Z., Jordan, M. I., 1997. Factorial hidden Markov models. Machine Learning 29 (2-3), 245–273.

Haft, M., Hofmann, R., Tresp, V., 1999. Model-independent mean-field theory as a local method for approximate propagation of information. Network: Computation in Neural Systems 10, 93–105.

Heskes, T., Albers, K., Kappen, B., 2003. Approximate inference and constrained optimization. In: UAI. pp. 313–320.

Jaakkola, T., 2001. Tutorial on variational approximation methods. In: Saad, D., Opper, M. (Eds.), Advanced Mean Field Methods. MIT Press, Ch. 10, pp. 129–160.

Jensen, F. V., Lauritzen, S., Olesen, K., 1990. Bayesian updating in recursive graphical models by local computation. Computational Statistics Quarterly 4, 269–282.

Jordan, M. I., Ghahramani, Z., Jaakkola, T., Saul, L. K., 1999. An introduction to variational methods for graphical models. Machine Learning 37 (2), 183–233.

Kjærulff, U., 1994. Reduction of computational complexity in Bayesian networks through removal of weak dependences. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence. pp. 374–382.

Lauritzen, S. L., Spiegelhalter, D. J., 1988. Local computations with probabilities on graphical structures and their application to expert systems. Journal of Royal Statistics Society, Series B 50 (2), 157–224.

McEliece, R. J., MacKay, D. J. C., Cheng, J.-F., 1998. Turbo decoding as an

instance of Pearl's "belief propagation" algorithm. IEEE Journal on Selected Areas in Communications 16 (2), 140–152.

Minka, T. P., 2001. A family of algorithms for approximate Bayesian inference. Ph.D. thesis, MIT.

Minka, T. P., Qi, Y. A., 2003. Tree-structured approximations by expectation propagation. In: Advances in Neural Information Processing Systems (NIPS). pp. 193–200.

Mooij, J. M., 2008. libDAI 0.2.2: A free/open source C++ library for discrete approximate inference methods. http://mloss.org/software/view/77/.

Murphy, K. P., Weiss, Y., Jordan, M. I., 1999. Loopy belief propagation for approximate inference: An empirical study. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence. pp. 467–475.

Park, J., Darwiche, A., 2004. A differential semantics for jointree algorithms. Artificial Intelligence 156, 197–216.

Pearl, J., 1988. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann Publishers, Inc., San Mateo, California.

Saul, L. K., Jordan, M. I., 1995. Exploiting tractable substructures in intractable networks. In: Advances in Neural Information Processing Systems (NIPS). pp. 486–492.

Suermondt, H. J., 1992. Explanation in bayesian belief networks. Ph.D. thesis, Stanford.

Sun, J., Li, Y., Kang, S. B., 2005. Symmetric stereo matching for occlusion handling. In: CVPR (2). pp. 399–406.

Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M. F., Rother, C., 2006. A comparative study of energy minimization methods for Markov random fields. In: ECCV (2). pp. 16–29.

van Engelen, R. A., 1997. Approximating Bayesian belief networks by arc removal. IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (8), 916–920.

Wainwright, M. J., Jaakkola, T., Willsky, A. S., 2003. Tree-based reparameterization framework for analysis of sum-product and related algorithms. IEEE Transactions on Information Theory 49 (5), 1120–1146.

Weiss, Y., 2001. Comparing the mean field method and belief propagation for approximate inference in MRFs. In: Saad, D., Opper, M. (Eds.), Advanced Mean Field Methods. MIT Press, Ch. 15, pp. 229–239.

Welling, M., 2004. On the choice of regions for generalized belief propagation. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence. AUAI Press, Arlington, Virginia, p. 585.

Welling, M., Minka, T. P., Teh, Y. W., 2005. Structured region graphs: Morphing EP into GBP. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence. pp. 609–614.

Wiegerinck, W., 2000. Variational approximations between mean field theory and the junction tree algorithm. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence. pp. 626–633.

Xing, E. P., Jordan, M. I., Russell, S. J., 2003. A generalized mean field algorithm for variational inference in exponential families. In: UAI. pp. 583–591.

Yedidia, J., Freeman, W., Weiss, Y., 2005. Constructing free-energy approximations and generalized belief propagation algorithms. IEEE Transactions on Information Theory 51 (7), 2282–2312.

Yedidia, J. S., Freeman, W. T., Weiss, Y., 2003. Understanding belief propagation and its generalizations. In: Lakemeyer, G., Nebel, B. (Eds.), Exploring Artificial Intelligence in the New Millennium. Morgan Kaufmann, Ch. 8, pp. 239–269.

Zhang, N. L., Poole, D., 1996. Exploiting causal independence in bayesian network inference. Journal of Artificial Intelligence Research 5, 301–328.