MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Characterization of belief propagation and its generalizations

Jonathan S. Yedidia, William T. Freeman, Yair Weiss

TR2001-15 March 2001

Abstract

Graphical models are used in many scientific disciplines, including statistical physics, machine learning, and error-correcting coding. One typically seeks the marginal probability of selected variables (nodes of the graph) given observations at other variables. Belief propagation (BP) is a fast marginalization method, based on passing local messages. Designed for singly-connected graphs, BP nonetheless works well in many applications involving graphs with loops, for reasons that were not well understood. We characterize the BP solutions, showing that BP can only converge to a stationary point of an approximate free energy, known as the Bethe free energy in statistical physics. This understanding lets us for construct new message-passing algorithms based on improvements to Bethe's approximation introduced by Kikuchi and others. The new generalized belief propagation (GBP) algorithms are much more accurate than ordinary BP for some problems, and permit solutions to Kikuchi approximations for otherwise intractable inhomogeneous systems. We illustrate GBP with a spin-glass example and an error-correcting code, showing dramatically improved estimates of local magnetizations and decoding performance using GBP.

Publication History:–

1. First printing, TR-2001-15, April 2001

Characterization of belief propagation and its generalizations

Jonathan S. Yedidia and William T. Freeman
Mitsubishi Electric Research Laboratories (MERL)
201 Broadway
Cambridge, MA 02139
yedidia, freeman@merl.com

Yair Weiss
Computer Science Division
UC Berkeley, 485 Soda Hall
Berkeley, CA 94720-1776
yweiss@cs.berkeley.edu

March 20, 2001

Abstract

Graphical models are used in many scientific disciplines, including statistical physics, machine learning, and error-correcting coding. One typically seeks the marginal probability of selected variables (nodes of the graph) given observations at other variables. Belief propagation (BP) is a fast marginalization method, based on passing local messages. Designed for singly-connected graphs, BP nonetheless works well in many applications involving graphs with loops, for reasons that were not well understood.

We characterize the BP solutions, showing that BP can only converge to a stationary point of an approximate free energy, known as the Bethe free energy in statistical physics. This understanding lets us construct new message-passing algorithms based on improvements to Bethe's approximation introduced by Kikuchi and others [1, 2]. The new generalized belief propagation (GBP) algorithms are much more accurate than ordinary BP for some problems, and permit efficient solutions to Kikuchi approximations for inhomogeneous systems. We illustrate GBP with a spin-glass example and an error-correcting code, showing dramatically improved estimates of local magnetizations and decoding performance using GBP.

Recent developments in the area of *graphical models* are having significant impact in many areas of computer science, engineering and applied mathematics. Graphical models, referred to in various guises as “Bayesian networks,” “Markov random fields,” “factor graphs,” or “structured stochastic systems,” are an elegant marriage of graph theory, probability theory and decision theory. They provide a system-theoretic, computational framework in which to do probability and statistics, justifying and unifying many classical techniques for inference, learning, and decision-making, and providing a firm foundation on which to design new systems. [3]

Figure 1 shows a simple example of an undirected graphical model. Each circle represents a random variable or an observation and the structure of the graph denotes the joint statistical dependence between the variables.

We may seek the marginal probability at a node. This could correspond to the local magnetization of a spin in a magnetic spin model, the probability of a particular object shape explaining an image for a vision problem, or the optimal decoding of a bit in an error-correcting code.

For many problems of practical size, direct marginalization is intractable. Monte Carlo methods are sometimes too slow while variational methods [4] require discovering good approximating functionals.

Belief propagation (BP) is an appealing local message-passing algorithm which is guaranteed to converge to the correct marginal probabilities in singly-connected graphs [5]. However, for general graphs with loops, the situation is less clear. For some problems, including decoding Turbo codes and some computer vision problems, BP works very well [6, 7, 8, 9]. For other graphs with loops, BP may give poor results or fail to converge [9]. Little has been understood about what approximation BP represents for a general graph, or how it might be improved.

Here we provide such an understanding, and introduce a set of new and more accurate inference algorithms for graphical models. We show that BP is the first in a progression of local message-passing algorithms, each giving equivalent results to a corresponding approximation from statistical physics known as the "Kikuchi" approximation to the free energy. The new algorithms can be much more accurate than ordinary BP, and may converge when ordinary BP does not.

For the purposes of exposition, we concentrate on undirected graphs with pairwise potentials (compatibility functions); the generalization to higher-order tensorial interactions is straightforward. The state of each node i is denoted by x_i, and the joint probability is

$$P(x_1, x_2, \ldots, x_N) = \alpha \prod_{ij} \psi_{ij}(x_i, x_j) \prod_i \psi_i(x_i) \tag{1}$$

where $\psi_i(x_i)$ is the local "evidence" for node i [10], $\psi_{ij}(x_i, x_j)$ is the compatibility matrix between nodes i and j, and α is a normalization constant.

The standard BP updates for the messages and the marginal probabilities as a function of other messages are:

$$b_i(x_i) \leftarrow \alpha\psi_i(x_i) \prod_{k \in N(i)} m_{ki}(x_i) \tag{2}$$

$$m_{ij}(x_j) \leftarrow \alpha \sum_{x_i} \psi_{ij}(x_i, x_j)\psi_i(x_i) \prod_{k \in N(i)\backslash j} m_{ki}(x_i) \tag{3}$$

where α denotes a normalization constant and $N(i)\backslash j$ means all nodes neighboring node i, except j. Here $m_{ij}(x_j)$ is the message that node i sends to node j and $b_i(x_i)$ is the belief (approximate marginal probability) at node i, obtained by multiplying all incoming messages to that node by the local evidence. Similarly, we can define the belief $b_{ij}(x_i, x_j)$ at the pair of nodes (x_i, x_j) as the product of the local potentials and all messages incoming to the pair of nodes: $b_{ij}(x_i, x_j) = \alpha\phi_{ij}(x_i, x_j) \prod_{k \in N(i)\backslash j} m_{ki}(x_i) \prod_{l \in N(j)\backslash i} m_{lj}(x_j)$, where $\phi_{ij}(x_i, x_j) \equiv \psi_{ij}(x_i, x_j)\psi_i(x_i)\psi_j(x_j)$.

The beliefs are subject to the normalization and marginalization constraints: $\sum_{x_i} b_i(x_i) = 1$, $\sum_{x_i} b_{ij}(x_i, x_j) = b_j(x_j)$. Figure 2 (a)-(c) shows graphically how the marginalization constraints combined with the belief equations lead to the BP message update rules.

Claim 1: Let $\{m_{ij}\}$ be a set of BP messages and let $\{b_{ij}, b_i\}$ be the beliefs calculated from those messages. Then the beliefs are fixed-points of the BP algorithm if and only if they are zero gradient points of the Bethe free energy, F_β [11, 12]:

$$F_\beta(\{b_{ij}, b_i\}) = \sum_{ij} \sum_{x_i, x_j} b_{ij}(x_i, x_j) \ln \frac{b_{ij}(x_i, x_j)}{\phi_{ij}(x_i, x_j)} - \sum_i (q_i - 1) \sum_{x_i} b_i(x_i) \ln \frac{b_i(x_i)}{\psi_i(x_i)} \tag{4}$$

(q_i is the number of neighbors of node i.)

The fact that $F_\beta(\{b_{ij}, b_i\})$ is bounded below implies that the BP equations always possess a fixed-point (obtained at the global minimum of F). To our knowledge, this is the first proof of existence of fixed-points for a general graph with arbitrary potentials (see [13] for a proof for a special case). The free energy formulation clarifies the relationship of BP to variational approaches, which also minimize an approximate free energy. [14, 4, 15].

The Bethe approximation, for which the energy and entropy are approximated by terms that involve at most pairs of nodes, is the simplest version of the Kikuchi "cluster variational method" [1, 2]. In a general Kikuchi approximation, the free energy is approximated as a sum of the free energies of basic clusters of nodes, minus the free energy of over-counted cluster intersections, minus the free energy of the over-counted intersections of intersections, and so on.

The choice of basic clusters determines the Kikuchi approximation–for the Bethe approximation, the basic clusters consist of all linked pairs of nodes. Let x_r be the state of the nodes in region r and $b_r(x_r)$ be the "belief" in x_r. We define the potential of a region by $\psi_r(x_r)$ by: $\psi_r(x_r) = \prod_{ij} \psi_{ij}(x_i, x_j) \prod_i \psi_i(x_i)$ where the products are over all interactions contained within the region r. For models with higher than pair-wise interactions, the region potential is generalized to include those interactions as well.

The Kikuchi free energy is

$$F_K = \sum_{r \in R} c_r \sum_{x_r} b_r(x_r) \ln \frac{b_r(x_r)}{\psi_r(x_r)} \tag{5}$$

where c_r is the over-counting number of region r, defined by: $c_r = 1 - \sum_{s \in super(r)} c_s$ where $super(r)$ is the set of all super-regions of r. We have $c_r = 1$ for the largest regions in the set of all regions, R. The belief $b_r(\alpha_r)$ in region r has several constraints: it must sum to one and be consistent with the beliefs in regions which intersect with r. In general, increasing the size of the basic clusters improves the approximation one obtains by minimizing the Kikuchi free energy.

Just as the Bethe free energy can be minimized by the BP algorithm, we introduce a class of analogous *generalized belief propagation* (GBP) algorithms that minimize an arbitrary Kikuchi free energy. There are in fact many possible GBP algorithms which all correspond to the same Kikuchi approximation. We present a "canonical" GBP algorithm which has the nice property of reducing to ordinary BP at the Bethe level. We introduce messages $m_{rs}(x_s)$ between all regions r and their "direct sub-regions" s [16]. It is helpful to think of this as a message from those nodes in r but not in s (which we denote by $r \backslash s$) to the nodes in s. Intuitively, we want messages to propagate information that lies outside of a region into it. Thus, for a given region r, we want the belief $b_r(x_r)$ to depend on exactly those messages $m_{r's'}$ that start outside of the region r and go into the region r. We define this set of messages $M(r)$ to be those messages $m_{r's'}(x_{s'})$ such that region $r' \backslash s'$ has no nodes in common with region r, and such that region s' is a sub-region of r or the same as region r.

The canonical GBP update rules are:

$$b_r(x_r) \leftarrow \alpha\psi_r(x_r) \prod_{m_{r's'} \in M(r)} m_{r's'}(x_{s'}) \quad (6)$$

$$b_s(x_s) = \sum_{x_{r\setminus s}} b_r(x_r) \quad (7)$$

The marginalization constraint, eq. (7) implicitly defines message update rules. Figure 2 (d)-(h) shows graphically how the marginalization constraints over the region probabilities leads to the message-passing equations for a particular choice of Kikuchi clusters.

Many different message-update schedules are possible, but it is a good idea to update the messages into the smallest regions first. One can then use the newly computed messages in the product over messages into larger regions. Empirically, this helps convergence.

Claim 2: Let $\{m_{rs}(x_s)\}$ be a set of canonical GBP messages and let $\{b_r(x_r)\}$ be the beliefs calculated from those messages. Then the beliefs are fixed-points of the canonical GBP algorithm if and only if they are zero gradient points of the constrained Kikuchi free energy F_K [17, 18].

We use the new algorithms in two well-understood problems. Figure 3 shows GBP used to calculate the local magnetizations of a row of spins in a spin glass [19]. This is equivalent to performing inference in a 2-d Markov random field, as used in computer vision problems [20]. BP works poorly for this example, and Monte Carlo simulations are very slow. The new GBP algorithm results are fast and nearly exact. This example illustrates how the GBP algorithm improves over the standard Bethe/TAP [21] approach to disordered spin systems from statistical physics, by providing an efficient way to minimize Kikuchi free energies for inhomogeneous systems on realistic lattices.

BP decoding works spectacularly well for Turbo Codes and Gallager codes. [7] Most explorations for new codes are now focused on those that can be represented by graphs without small loops, to insure that the BP algorithm will work well. But this restriction may prevent us from finding good codes which do have short loops in their graphical representations. Figure 4 is a graphical representation of a small toy (20,8,6) parity check code with many short loops (inspired by [22]). GBP decodes this code significantly better (by around a decibel or more depending on the cluster choice) than ordinary BP, illustrating that the GBP algorithm can be used to decode many codes that BP could not decode well. This code is sufficiently small so that the BP and GBP algorithms could be compared with exact (maximum likelihood) methods. When we decoded other parity check codes with short loops in their graphical representations that were too large to be tractable by maximum likelihood decoding, we obtained a qualitatively similar improvement in decoding using GBP compared BP. GBP algorithms should have many other applications in other science and engineering applications that involve inference in graphical models.

Footnotes

[1] R. Kikuchi. *Phys. Rev.*, 81:988, 1951.

[2] Special issue on Kikuchi methods. Progr. Theor. Phys. Suppl., vol. 115, 1994.

[3] M. I. Jordan, editor. *Learning in graphical models*. MIT Press, 1998.

[4] M.I. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul. An introduction to variational methods for graphical models. In M.I. Jordan, editor, *Learning in Graphical Models*. MIT Press, 1998.

[5] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 1988.

[6] B. J. Frey. *Graphical Models for Machine Learning and Digital Communication*. MIT Press, 1998.

[7] R.J. McEliece, D.J.C. MacKay, and J.F. Cheng. Turbo decoding as an instance of

Pearl's 'belief propagation' algorithm. *IEEE J. on Sel. Areas in Comm.*, 16(2):140–152, 1998.

[8] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael. Learning low-level vision. *Intl. J. Computer Vision*, 40(1):25–47, 2000.

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

[10] Note that we are subsuming any fixed evidence nodes into our definition of $\psi_i(x_i)$.

[11] Proof of claim 1: we add Lagrange multipliers to form a Lagrangian L: $\lambda_{ij}(x_j)$ is the multiplier corresponding to the constraint that $b_{ij}(x_i, x_j)$ marginalizes down to $b_j(x_j)$, and γ_{ij}, γ_i are multipliers corresponding to the normalization constraints. The equation $\frac{\partial L}{\partial b_{ij}(x_i,x_j)} = 0$ gives: $\ln b_{ij}(x_i, x_j) = \ln(\phi_{ij}(x_i, x_j)) + \lambda_{ij}(x_j) + \lambda_{ji}(x_i) + \gamma_{ij} - 1$. The equation $\frac{\partial L}{\partial b_i(x_i)} = 0$ gives: $(q_i - 1)(\ln b_i(x_i) + 1) = \ln \psi_i(x_i) + \sum_{j \in N(i)} \lambda_{ji}(x_i) + \gamma_i$. Setting $\lambda_{ij}(x_j) = \ln \prod_{k \in N(j) \setminus i} m_{kj}(x_j)$ and using the marginalization constraints, we find that the stationary conditions on the Lagrangian are equivalent to the BP fixed-point conditions. (Empirically, we find that *stable* BP fixed-points correspond to local *minima* of the Bethe free energy, rather than maxima or saddle-points.) Kabashima and Saad [24] have previously pointed out the correspondence between BP and the Bethe approximation for some specific graphical models with random disorder.

[12] Bayesian networks are directed graphs in which the conditional probability over all hidden nodes is:

$$P(x|y) = \frac{1}{Z} \prod_i P(x_i|\mathrm{Par}(x_i)) P(y_i|x_i), \tag{8}$$

where $\mathrm{Par}(x_i)$ means the parents of x_i, and we have assumed that every hidden node x_i has an observed node y_i connected to it. Then the Bethe free energy is:

$$F_\beta = -\sum_i \sum_{f_i} b(f_i) \ln P(x_i|\mathrm{Par}(x_i)) - \sum_i \sum_{x_i} b(x_i) \ln P(y_i|x_i)$$

$$+\sum_i \sum_{f_i} b(f_i) \ln b(f_i) - \sum_i (q_i - 1) \sum_{x_i} b(x_i) \ln b(x_i) \quad (9)$$

where f_i denotes the state of the "family" of x_i: $f_i = (x_i, \mathrm{Par}(x_i))$ and q_i is the number of families a node participates in.

[13] T. J. Richardson. The geometry of turbo-decoding dynamics. *IEEE Trans. Info. Theory*, 46(1):9–23, Jan. 2000.

[14] The BP free energy is a function of one-node beliefs $b_i(x_i)$ as well as two-node beliefs $b_{ij}(x_i, x_j)$, while the variational mean field free energy, F_{MF}, ordinarily depends only the one-node beliefs. It is easy to show that the BP free energy is exact for trees while the mean field one is not. Furthermore the optimization methods are different: typically F_{MF} is minimized directly in the primal variables $\{b_i\}$ while F_β is minimized using the messages, which are a combination of the dual variables $\{\lambda_{ij}(x_j)\}$.

[15] D. J. C. MacKay. Free energy minimization algorithm for decoding and cryptanalysis. *Electronics Letters*, 31(6):446–447, 1995.

[16] Define the set $sub_d(r)$ of direct sub-regions of r to be those regions that are sub-regions of r but have no super-regions that are also sub-regions of r, and similarly for the set $super_d(r)$ of "direct super-regions."

[17] Outline of proof of claim 2: We add Lagrange multipliers: γ_r to enforce the normalization of b_r and $\lambda_{rs}(x_s)$ to enforce the consistency of each region r with all of its direct sub-regions s. We then rotate to another set of Lagrange multipliers $\mu_{rs}(x_s)$ of equal dimensionality which enforce a linear combination of the original constraints: $\mu_{rs}(x_s)$ enforces all those constraints involving marginalizations by all direct super-regions r' of s into s except that of region r itself. The rotation matrix is in a block form which can be guaranteed to be full rank. We can then show that the $\mu_{rs}(x_s)$ constraints can be written in the form $\mu_{rs}(x_s) \sum_{r' \in R(\mu_{rs})} c_{r'} \sum_{x_{r'}} b(x'_r)$ where $R(\mu_{rs})$ is the set of all regions which receive the mes-

sage μ_{rs} in the belief update rule of the canonical algorithm. We re-arrange the sum over all μ's into a sum over all regions, which has the form $\sum_{r \in R} c_r \sum_{x_r} b_r(x_r) \sum_{\mu_{rs} \in \tilde{M}(r)} \mu_{rs}(x_s)$. ($\tilde{M}(r)$ is a set of $\mu_{r's'}$ in one-to-one correspondence with the $m_{r's'}$ in $M(r)$.) Finally, we differentiate the Kikuchi free energy with respect to $b_r(r)$, and identify $\mu_{rs}(x_s) = \ln m_{rs}(x_s)$ to obtain the canonical GBP belief update rules, Eq. 6.

Other GBP message passing algorithms are also equivalent to the Kikuchi approximation. If one writes any set of constraints which are sufficient to insure the consistency of all Kikuchi regions, one can associate the exponentiated Lagrange multipliers of those constraints with a set of messages.

[18] The GBP algorithms we have described solve exactly those graphs which have the topology of a tree of basic clusters. This is reminiscent of Pearl's method of clustering [5], wherein one groups clusters of nodes into "super-nodes," and then applies a belief propagation method to the equivalent super-node graph. We can show that the clustering method, using Kikuchi clusters as super-nodes, gives results equivalent to the Kikuchi approximation when the over-counting number $c_r \leq 0$ for all intersection sub-regions. For those graphs and cluster choices which do not obey this condition Pearl's clustering method must be modified by adding additional update conditions to agree with the GBP algorithm and the Kikuchi approximation.

[19] M. Mezard, G. Parisi, and M. A. Virasoro. *Spin glass theory and beyond*. World Scientific, 1987.

[20] S. Geman and D. Geman. Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images. *IEEE Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

[21] D. J. Thouless, P. W. Anderson, and R. G. Palmer. Solution of 'solvable model of a spin glass'. *Phil. Mag.*, 35:593–601, 1977.

[22] D. J. C. MacKay. *Information Theory, Inference and Learning Algorithms*
. http://wol.ra.phy.cam.ac.uk/mackay/Book.html, 2000. Figure 1.18.

[23] R. G. Gallager. *Low-density parity check codes*, volume 21. MIT Press, 1963.

[24] Y. Kabashima and D. Saad. Belief propagation vs. TAP for decoding corrupted messages. *Euro. Phys. Lett.*, 44:668, 1998.

We thank D. Forney and M. Jordan for helpful discussions.

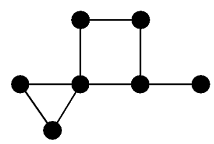


Figure 1: Example undirected graphical model. Graph nodes represent random variables, which may be continuous or discrete. Lines indicate statistical dependencies. We typically want to find the marginal probability at a node.

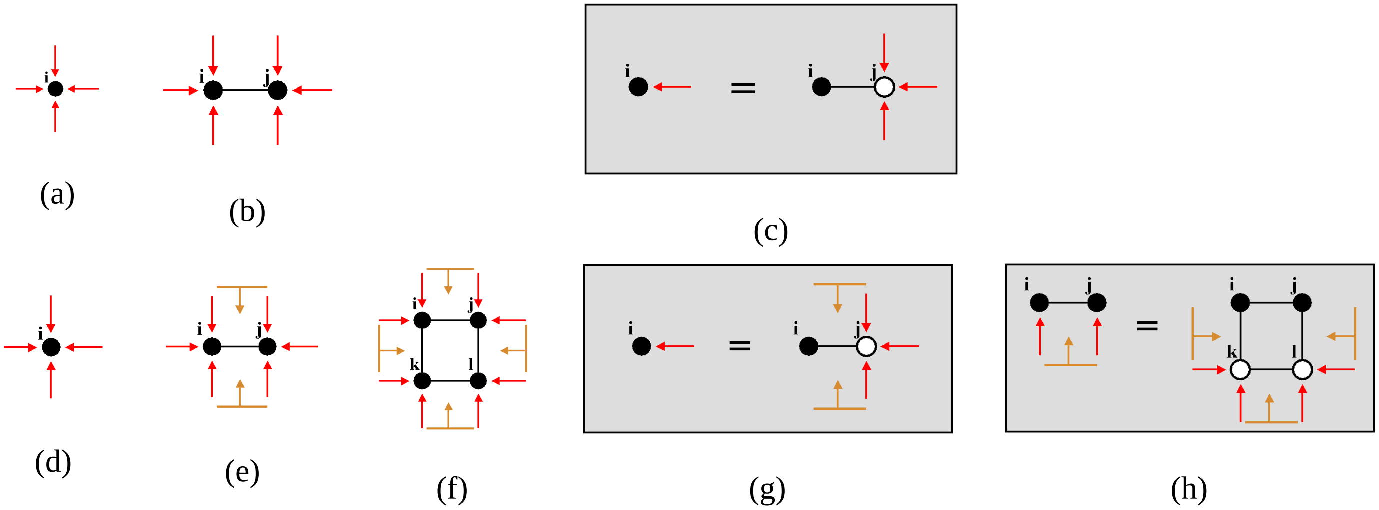
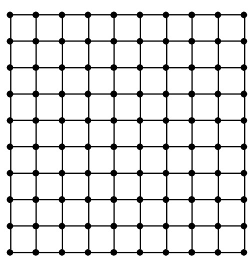
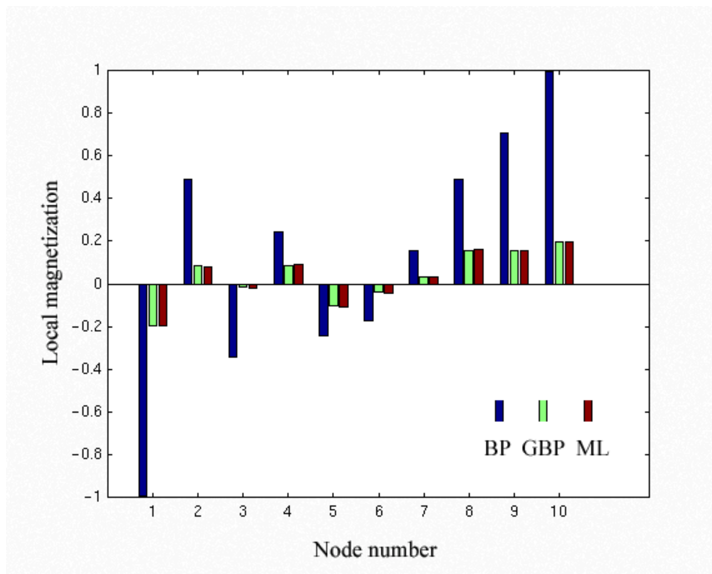


Figure 2: Marginal probabilities (beliefs) and associated message update equations for BP and GBP on a grid Markov random field, such as Fig. 3 (a). **BP** Illustration of Eq. (2) for the belief at node i: $b_i(x_i) \propto \psi_i(x_i) m_{ji}(x_i) m_{ki}(x_i) m_{li}(x_i) m_{mi}(x_i)$, where j, k, m, and n are the neighboring nodes; $m_{ai}(x_i)$ is a message from node a to node i; and $\psi_i(x_i)$ is the local evidence at node i. (b) Belief at nodes i and j, $b_{ij}(x_i, x_j) \propto \psi_i(x_i)\psi_j(x_j)\psi_{ij}(x_i, x_j) m_{ki}(x_i) m_{li}(x_i) m_{mi}(x_i) m_{nj}(x_j) m_{oj}(x_j) m_{pj}(x_j)$, where n, o, and p are neighbors to node j. Requiring that $b_{ij}(x_i, x_j)$ marginalize down to $b_i(x_i)$ gives the BP message update equation, Eq. (2), shown in (c). Open circle indicates a node which is marginalized over. **GBP**, using 4-node Kikuchi clusters. Depiction of the beliefs at a node (d), pair of nodes (e), and at a cluster of 4 nodes (f) in terms of GBP messages. These are instances of Eq. (6). Red arrows are the single index messages also used in ordinary BP. Brown segments indicate the double-indexed messages of this GBP approximation, corrections to the BP messages. The requirement that (e) marginalize down to (d) leads to one GBP update equation, (g); that (f) must marginalize down to (e) leads to the other, (h). Both are instances of message update equations derived using Eq. (7). Fixed points of these message update equations give beliefs that are stationary points (empirically minima) of the corresponding Kikuchi approximation to the free energy.



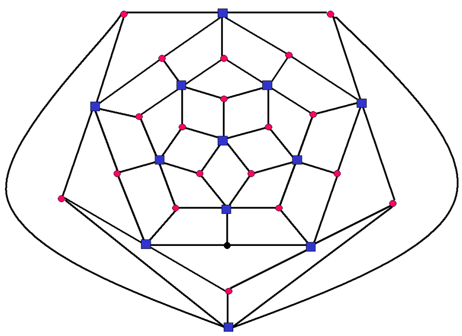
(a)



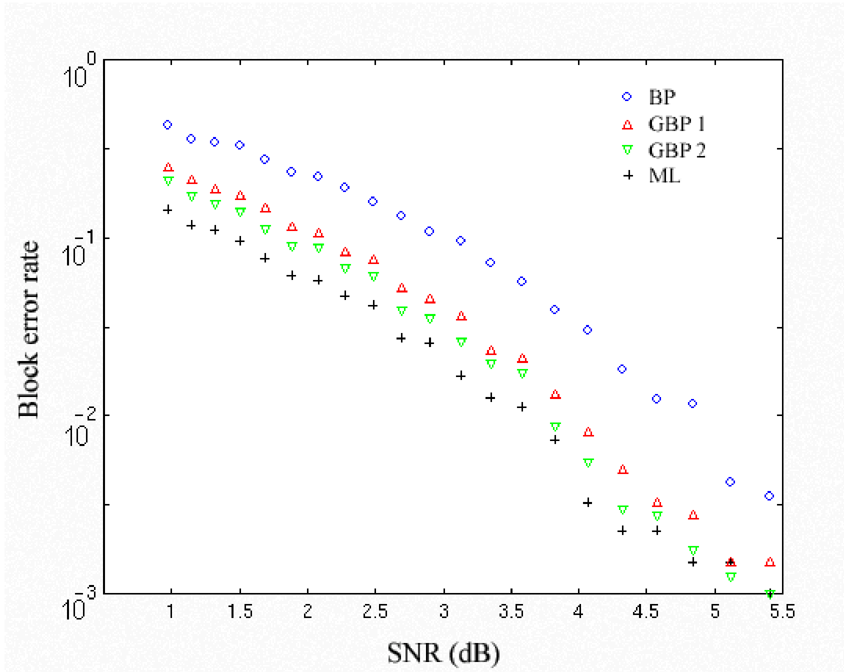
(b)

Figure 3: Caption on next page.

Figure 3: Solution of square lattice Ising spin glass in a random magnetic field using GBP. Nearest neighbor nodes connected by a compatibility matrix of the form $\psi_{ij} = \begin{pmatrix} \exp(J_{ij}) & \exp(-J_{ij}) \\ \exp(-J_{ij}) & \exp(J_{ij}) \end{pmatrix}$ and local evidence vectors of the form $\psi_i = (\exp(h_i); \exp(-h_i))$. Similar models are used in computer vision and image processing problems [20]. To instantiate a particular lattice, the J_{ij} and h_i parameters are chosen randomly and independently from zero-mean Gaussian probability distributions with standard deviations J and h respectively. The following results are for n by n lattices with toroidal boundary conditions and with $J = 1$, and $h = 0.1$. This model, with many small loops, conflicting interactions, and weak evidence, shows the weaknesses of ordinary BP, which performs well for many other networks. We started with randomized messages and only stepped half-way towards the computed values of the messages at each iteration in order to help convergence. We found that canonical GBP took about twice as long as ordinary BP per iteration, but would typically reach a given level of convergence in many fewer iterations. In fact, for the majority of the dozens of samples that we looked at, BP did not converge at all, while canonical GBP always converged for this model and always to accurate answers. (We found that for the zero-field 3-dimensional spin glass with toroidal boundary conditions, canonical GBP with 2x2x2 cubic clusters would also fail to converge). For $n = 20$ or larger, it was difficult to make comparisons with any other algorithm, because ordinary BP did not converge and Monte Carlo simulations suffered from extremely slow equilibration. However, generalized belief propagation converged reasonably rapidly to plausible-looking beliefs. For small n, we could compare with exact results, found by clustering sets of n nodes into a chain. For the $n = 10$ lattice shown in (a), BP did converge and we compare BP, GBP, and the exact local magnetizations for one row of the spins, plotted in (b). BP seriously exaggerates the magnetizations, while GBP finds them nearly exactly.



(a)



(b)

Figure 4: Graphical representation of a (20,8,6) parity check code. The code is obtained from a dodecahedron–each corner of the dodecahedron (represented by a red circle) is a bit in the code and each face of the dodecahedron (represented by a blue square) is a parity check of the bits on the corners of that face. This code has a reasonably large minimum distance of 6 bits between code words, but has small loops in its graphical representation. We used the additive white Gaussian noise channel, and monitored the percentage of successfully decoded code words (over 4000 transmitted words) at various noise rates. We compare the results of four decoding methods: maximum likelihood (for such a small code table-lookup is tractable), BP (Gallager decoding [23]), GBP using clusters of the five nodes included in a single parity check, and GBP using clusters of the ten nodes included in three adjacent parity checks. The BP and GBP algorithms were each run for 100 iterations for each block, and the GBP messages were updated so as to move half-way to their updated values at each iteration. We find that GBP decoding, even using the small clusters of 5 nodes, improves the decoding performance by nearly a decibel in terms of the signal to noise ratio necessary to achieve a given decoding success rate.