Efficient MAP approximation for dense energy functions

Marius Leordeanu MLEORDEA@ANDREW.CMU.EDU
Martial Hebert HEBERT@RI.CMU.EDU
The Robotics Institute, Carnegie Mellon University, Pennsylvania, PA

Abstract

We present an efficient method for maximizing energy functions with first and second order potentials, suitable for MAP labeling estimation problems that arise in undirected graphical models. Our approach is to relax the integer constraints on the solution in two steps. First we efficiently obtain the relaxed global optimum following a procedure similar to the iterative power method for finding the largest eigenvector of a matrix. Next, we map the relaxed optimum on a simplex and show that the new energy obtained has a certain optimal bound. Starting from this energy we follow an efficient coordinate ascent procedure that is guaranteed to increase the energy at every step and converge to a solution that obeys the initial integral constraints. We also present a sufficient condition for ascent procedures that guarantees the increase in energy at every step.

1. Introduction

Efficient methods for MAP inference in graphical models are of major interest in pattern recognition, computer vision and machine learning. The MAP problem often reduces to optimizing an energy function that depends on how well the labels agree with the data (first-order potentials) as well as on how well pairs of labels at connected sites agree with each other (second order potentials). We propose an efficient method for approximately maximizing such energy functions, without imposing any constraints on the first or second order terms. Our method converges to a solution that satisfies an optimality bound that is data independent.

Graph cuts have been successful in labeling tasks with regular energy functions, especially those related to low level vision (Boykov, Veskler, Rabih, 2001). For binary labeling problems graph cuts are provably optimal. For multiple labels, the optimality bound given in Boykov, Veskler and Rabih (2001) is data dependent and for arbitrary energy functions it could be arbitrarily far from the optimum. In contrast, our approach works with general energy functions.

Loopy Belief Propagation and its variants (e.g. Tree Re-weighted Belief Propagation, closely related to Linear Programming Relaxation (Wainwright 2005)), have also shown experimental success. The correctness and convergence of Loopy BP is not guaranteed for general graphs and energy functions and in some cases it does not converge to good approximations (Murphy, Weiss and Jordan, 1999).

Other iterative optimization techniques for labeling classification problems such as deterministic annealing, self-annealing, self-annihilation (Rangarajan, 2000) or relaxation labeling (Hummel and Zucker, 1993) are shown to improve the energy at each iteration (Yuille and Rangarajan, 2003), but there is no result as far as we know regarding their optimality properties.

Our work is related to the optimization of polynomials with nonnegative coefficients under relaxed $L2$ constraints (Baratchart ,Berthod, Pottier, 1998). Our algorithm has two stages. In the first stage we follow a path similar to the one from Baratchart (1998) by relaxing the constraints on the solution (Section 3) and obtaining the exact optimum for the relaxed problem. In the second stage we show how to iteratively increase the energy at every step (not necessarily strictly), and how to obtain a solution respecting the original constraints, which is also guaranteed to be close to the optimum (Section 4).

We are particularly interested in energy functions for arbitrary graphs with arbitrary number of labels. This is in contrast with recent studies of energy optimization tasks in computer vision, which are mainly focus-

Appearing in *Proceedings of the 23rd International Conference on Machine Learning*, Pittsburgh, PA, 2006. Copyright 2006 by the author(s)/owner(s).

ing on weakly connected graphs such as trees or planar graphs (Szeliski, 2006). More complex graphs with arbitrary second order potentials are important in higher level computer vision tasks, such as object recognition and scene analysis. Data dependent second order potentials are typically used in Conditional Random Fields (CRF) (Kumar, 2003; Quattoni, 2005). In object recognition problems, the nodes could correspond to different object parts, while the second order potentials would describe how those parts interact given the data. At this level of representation, neither simple second-order potentials such as in the Potts model, nor simpler graphs structures such as planar graphs would be appropriate.

2. Problem Formulation

We are addressing the problem of maximizing the energy functions that typically arise in labeling problems. The following form of the energy function follows previous work such as deterministic annealing (Rangarajan, 2000):

$$E = 1/2 \sum_{ia;jb} x_{ia} x_{jb} Q_{ia;jb} + \sum_{ia} x_{ia} D_{ia} \qquad (1)$$

Here $Q_{ia;jb}$ corresponds to the higher order term describing how well the label a at site i agrees with the label b at site j, given the data. $Q_{ia;jb}$ could also be a smoothness term independent of the data (such as the Potts model) that simply encourages neighboring sites to have similar labels. We set $Q_{ia;jb} = 0$ if $i = j$ or if the sites i and j are not connected since $Q_{ia;jb}$ should describe connections between different sites. For each pair of site i and its possible label a, the first order potentials are represented by D_{ia}, which in general describes how well the label a agrees with the data at site i.

In this formulation x is required to be an indicator vector with an entry for each pair of $(site, label)$, such that if $x_{ia} = 1$ site i is assigned label a and $x_{ia} = 0$ otherwise. Thus, with a slight abuse of notation, we consider ia to be a unique index given to the pair site i and label a. Also, one site can be given one and only one label. These constraints are enforced by requiring $x_{ia} \in \{0, 1\}$ and $\sum_a x_{ia} = 1$. The vector x will be of dimension $S * L$, where S is the number of sites and L the number of labels for each site. To simplify notations we assume that for each site there is an equal number of labels, but our approach can accommodate different number of labels for each site.

It is convenient to write the expression for E in the matrix form $E = 1/2x^T Qx + Dx$, where $Q(ia, jb) = Q_{ia;jb}$, $D(ia, ia) = D_{ia}$, and $Q(ia, ia) = 0$.

We assume that Q and D have non-negative elements, without loss of generality. That is true because we could change them to have non-negative elements, without changing the global optimum solution of the energy under the integral constraints. More specifically: let c be the smallest element in both Q and D (we assume c is finite). Let C_m be a matrix of the same size as Q, and C_v a vector of the same size as D, both with constant elements equal to c, except for the diagonal of C_m, which is set to zero . Due to the integral constraints on x, we have $1/2x^T C_m x + C_v x = \frac{c*S*(S-1)}{2} + cS$, which is independent of x. Therefore x^* that maximizes E also maximizes $1/2x^T(Q - C_m)x + (D - C_v)x$. Next, we can redefine Q as $Q - C_m$ and D as $D - C_v$, so that both have only non-negative elements. The non-negativity of Q also brings a practical advantage, especially when most of its smallest elements are very close to 0. Those elements could be set to 0 at the cost of a small change in the energy, with the benefit of a significant decrease in memory cost (since Q should be stored as a sparse matrix).

We attack the problem in two stages. During the first step, we relax the constraints to $\sum_a x_{ia}^2 = 1$, and find the global optimum of the polynomial with non-negative coefficients given by our energy function. The procedure is extremely fast, being very similar to the iterative power method for finding the largest eigenvector of the matrix Q and it usually converges in a few iterations (Section 3). Then we map the relaxed global optimum on the simplex given by $\sum_a x_{ia} = 1$ and show that the energy thus obtained is close to the global maximum (Section 3.1)

This gives us a good starting point for the second stage when we follow an iterative procedure that is guaranteed to increase the energy after every iteration (not necessarily strictly) and converge to a solution that obeys the initial integral constraints.

3. Global Optimum under Relaxed Constraints

We start by globally maximizing the energy function E under the relaxed constraints $\sum_a x_{ia}^2 = 1$. Introducing Lagrange multipliers we obtain the free energy as:

$$F = 1/2x^T Qx + Dx + \sum_i \lambda_i (\sum_a x_{ia}^2 - 1) \qquad (2)$$

Setting the partial derivatives with respect to x and the parameters λ_i to zero, and solving for the Lagrange

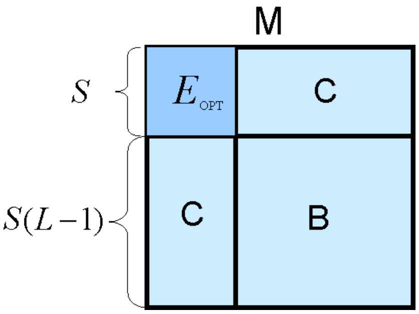

Figure 1. M has the second order potentials on its off diagonal elements and the first order elements on its diagonal

multipliers we obtain:

$$x^*_{ia} = \frac{\sum_{jb} Q_{ia;jb} x^*_{jb} + D_{ia}}{\sqrt{\sum_{b=1}^{L} {x^*_{ib}}^2}} \qquad (3)$$

This equation looks somehow similar to the eigenvector equation $Qx = \lambda x$ if it were not for the vector D and the site-wise normalization instead of the global one which applies to eigenvectors. Starting from a vector with positive elements, the fixed point x^* of the above equation has positive elements, is unique and it is a global maximum of the energy E under the constraints $\sum_a x_{ia}^2 = 1$, due to the fact that Q and D have non-negative elements (Theorem 5, Baratchart et al., 1998). Let E^* be the optimal energy $E(x^*)$, E_0 the energy after bringing x^* on the simplex $\sum_a x_{ia} = 1$, and E_{opt} the true optimal for the original integral constraints.

4. Mapping the Relaxed Solution on the Simplex

From vector x^* we can obtain a vector x_0 that lies on the simplex $\sum_a x_{ia} = 1$, by setting $x_{0ia} = x^*_{ia} / \sum_b x^*_{ib}$. Next we show that the energy E_0 evaluated at x_0 satisfies $E_0 \geq \frac{1}{L} E^* \geq \frac{1}{L} E_{opt}$, where L is the number of labels. This is a very loose lower bound, but it is important since it does not depend on the actual energy function.

We start by expressing E_0 in terms of x^*:

$$E_0 = \frac{1}{2} \sum_{ia;jb} \frac{x^*_{ia}}{\sum_b x^*_{ib}} \frac{x^*_{jb}}{\sum_b x^*_{jb}} Q_{ia;jb} + \sum_{ia} \frac{x^*_{ia}}{\sum_b x^*_{ib}} D_{ia} \qquad (4)$$

We know that x^* has non-negative elements and it satisfies $\sum_b {x^*}^2_{ib} = 1$, therefore we have $\sum_b x^*_{ib} \leq \sqrt{L}$.

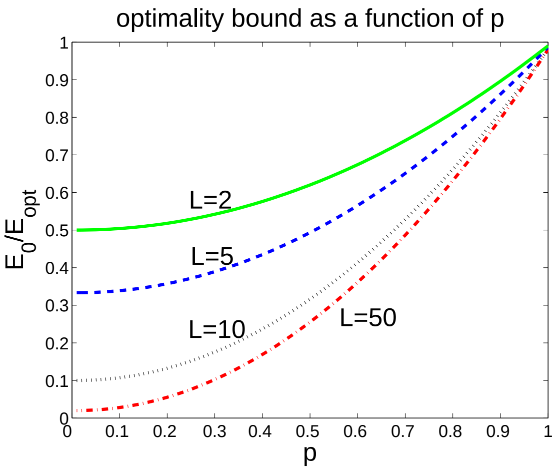

Figure 2. Optimality bound as we vary p, where p is the maximum $\in [0, 1]$ such that $B \geq p(L-1)^2 E_{opt}$ and $C \geq p(L-1)E_{opt}$

If we let $k = max_i(\sum_b x^*_{ib})$, it immediately follows that

$$E_0 \geq \frac{1}{k^2}\left(\frac{1}{2}\sum_{ia;jb} x^*_{ia} x^*_{jb} Q_{ia;jb} + \sum_{ia} x^*_{ia} D_{ia}\right) \geq \frac{1}{k^2} E^* \qquad (5)$$

We must also have $E_{opt} \leq E^*$, since any solution satisfying the original constraints also satisfy the relaxed ones, and E^* is the global optimum over the relaxed constraints. Thus we obtain $E_0 \geq \frac{1}{k^2} E_{opt}$, where $k \in [1, \sqrt{L}]$. Obviously this bound is very loose since we replaced the sums over the elements of x^* for each site by their largest possible value. However, the bound reflects the desirable property that the more peaked the elements in x^* are, the lower the sums $\sum_b x^*_{ib}$, and thus the closer E_0 will be to E_{opt}. One can find tighter bounds if the actual second and first order potentials are taken in consideration, as we will show next.

5. Data Dependent Lower Bound

For better understanding how far E_0 is from E_{opt} it is useful to define the matrix M such that $M = \frac{1}{2}Q + I_D$, where I_D is the diagonal matrix with $I_D(ia, ia) = D(ia)$. Then we have $M(ia, jb) = \frac{1}{2}Q(ia, jb)$ for any $i \neq j$ and $M(ia, ia) = D(ia)$. It follows that:

$$E(x) = 1/2x^T Qx + Dx \geq x^T Mx \qquad (6)$$

For clarity, let us assume, without loss of generality, that the elements of the optimal labeling x_{opt} have been permuted such that x_{opt} has ones on the first S elements and zeroes everywhere else, which implies the corresponding permutation of the rows and columns of M (Figure 1). Now the sum of all elements in the upper left S by S block of the matrix M is equal to the optimal energy E_{opt}. B and C are the sums over all elements in the corresponding sub-blocks of M as shown in Figure 1. Now let p be the maximum in $[0, 1]$, such that the average element in block B and the average element in block C are both greater or equal to p times the average element in E_{opt}. After considering the number of elements in B and C relative to E_{opt} it follows that $B \geq p(L-1)^2 E_{opt}$ and $C \geq p(L-1)E_{opt}$. Such a p always exists since B and C have only non-negative elements. We will show that the larger p is, the closer the energy E_0 is to the optimal energy E_{opt}:

Inequality 1: For any p as defined previously $E_0 \geq \frac{1+(L-1)p^2}{L} E_{opt}$

Proof: Let E_q be the the energy evaluated at a vector x constructed as a function of $q \geq 0$: $x_{ia} = \frac{1}{\sqrt{1+q^2(L-1)}}$ if a is an optimal assignment and $x_{ia} = \frac{q}{\sqrt{1+q^2(L-1)}}$ otherwise. Clearly, x satisfies the relaxed constraints $\sum_a x_{ia}^2 = 1$. Then (using inequality 6) we have:

$$E_q \geq x_q^T M x_q = \frac{E_{opt} + 2qC + q^2 B}{1 + q^2(L-1)} \qquad (7)$$

Since $B \geq p(L-1)^2 E_{opt}$ and $C \geq p(L-1)E_{opt}$ it follows that:

$$E_q \geq \frac{(1 + 2qp(L-1) + q^2 p(L-1)^2)E_{opt}}{1 + q^2(L-1)} \qquad (8)$$

Since $E_q \leq E^*$ and $p \in [0..1]$ we have

$$E^* \geq \frac{(1 + pq(L-1))^2}{1 + q^2(L-1)} E_{opt} \qquad (9)$$

The right hand side of the above inequality is maximized for $q = p$. Thus, if we use $E_0 \geq \frac{1}{L}E^*$ we have our bound:

$$E_0 \geq \frac{1 + (L-1)p^2}{L} E_{opt} \qquad (10)$$

The bound approaches 1 as p approaches 1 and it is always greater or equal to $1/L$. The curve becomes less and less sensitive to the number of labels as L becomes very large (Figure 2). Also, the lower the p the looser the bound. As future work, it would be interesting to explore tighter bounds depending on B and C relative to E_0. The less interesting result (easy to show) is that when both B and C approach zero, E_0 approaches E_{opt}.

In the next Section we show how, by starting from x_0 we follow a coordinate ascent approach in which we are guaranteed to increase the energy at every iteration until we reach a solution that satisfies the original constraints $\sum_a x_{ia} = 1$ and $x_{ia} \in \{0, 1\}$

6. Climbing Stage

Starting from an energy level E_0, which obeys an optimality bound, we continue through a special coordinate ascent procedure that guarantees both convergence and the satisfaction of the integral constraints. We show how any update rule that obeys inequality 2 is guaranteed to improve the energy at every step. We also give a general form of such update rules.

This stage is somehow similar to previous methods such a deterministic annealing, self-annealing, relaxation labeling and Iterative Conditional Modes (ICM). Among these only ICM is a coordinate ascent method. While all these methods could have arbitrary starting points, we start this stage at a point that is independent of our initial conditions, since it is the unique global optimum of the relaxed problem from the previous stage. Therefore, we can regard the previous stage as an initialization procedure, which is appropriate since it respects certain optimality bounds.

The algorithm outline is:

```
1. Stage 1: find $x^*$ that maximizes $1/2x^TQx + Dx$, given $\sum_a x_{ia}^2 = 1$ by iterating until convergence:
   (a) let $x = Qx + D$
   (b) normalize $x$ per site: $x_{ia} = \frac{x_{ia}}{\sqrt{\sum_b x_{ib}^2}}$
2. Initialize $x$, such that $x_{ia} = x_{ia}^* / \sum_b x_{ib}^*$
3. Stage 2: Set $\beta = 1$ and repeat until convergence
   (a) set $v_{ia} = \sum_{jb} Q_{ia;jb} x_{jb}^{(t)} + D_{ia}$
   (b) $x_{ia}^{(t+1)} = \sigma_i x_{ia}^{(t)} F(v_{ia}, \beta)$, where $\sigma_i = 1/\sum_b x_{ib}^{(t)} F(v_{ib}, \beta)$
   (c) increase $\beta$ after updating $x$ for all sites
```

We show that if the assignments at step 3.b are done site-wise (that is, vector x is updated sequentially) and $F(v, \beta)$ is any positive, monotonically increasing function of v, the energy increases at every site-wise update until convergence. If F is increasing exponentially with β then, by increasing β, we make the assignment at step 3.2 approach a *max* function such that x gets closer and closer to the original integral constraints. At the last step we actually set $x_{ia} = 1$ for which v_{ia} is maximum over all v_{ib}'s and 0 otherwise, thus satisfying the original integral constraints (the very last iteration is basically identical to ICM, also guaranteed to increase the energy).

Since every time we visit a site i we only update the values in vector x corresponding to that site, we can write the energy at that moment t as $E^{(t)} = E^{(t)}_{const} + E^{(t)}_i$, where $E^{(t)}_{const}$ is independent of x_{ia} for any label a and $E^{(t)}_i$ is a function of the variables x_{ia}. Therefore, it suffices to show that after updating the x_{ia}'s we increase the component E_i. One can show that $E^{(t)}_i$ can be expressed as:

$$E^{(t)}_i = \sum_a x^{(t)}_{ia} (\sum_{jb; j \neq i} x^{(t)}_{jb} Q_{ia;jb} + D_{ia}) = \sum_a x^{(t)}_{ia} v_{ia} \tag{11}$$

In the formula above the v_{ia}'s are independent of the x_{ia}'s, thus they do not change after the update. Therefore it is left to show that $\sum_a x^{(t)}_{ia} v_{ia} \leq \sum_a x^{(t+1)}_{ia} v_{ia} = \sum_a \sigma_i x^{(t)}_{ia} F(v_{ia}, \beta) v_{ia}$. The proof is based on the following inequality:

Inequality 2: Given any non-negative arrays b_q, w_q, and w^*_q, with $q = 1...n$ and $\sum_q w_q > 0$, $\sum_q w^*_q > 0$, such that $\frac{w^*_q}{w^*_k} \geq \frac{w_q}{w_k}$ whenever $b_q \geq b_k$, the following inequality holds: $\frac{\sum_q w^*_q b_q}{\sum_q w^*_q} \geq \frac{\sum_q w_q b_q}{\sum_q w_q}$

Proof: The proof relies on the fact (not proved here) that there must exists a k between 1 and n such that $\frac{w^*_q}{\sum_r w^*_r} \geq \frac{w_q}{\sum_r w_r}$ for any q such that $b_q \geq b_k$ and $\frac{w^*_q}{\sum_r w^*_r} \leq \frac{w_r}{\sum_q w_r}$ otherwise.

This inequality applies immediately to our problem if we set $w^*_a = x^{(t)}_{ia} F(v_{ia}, \beta)$, $w_a = x^{(t)}_{ia}$ and $b_a = v_{ia}$. In our experiments we used $F(v, \beta) = exp(\beta v)$. Other functions could be easily designed, such as $F(v, \beta) = \lambda + \gamma v^\beta$ with positive λ, γ (this is a generalization of the usual relaxation labeling update, also related to Baum and Sell, 1967). In experiments, both choices of F behaved very similarly.

7. Experimental Analysis

We compared our algorithm against other methods such as Max Product Loopy Belief Propagation (commonly used for MAP estimation), Deterministic Annealing (DA), Self-Annealing (SA), ICM, Deterministic Pseudo-Annealing (DPA) and Relaxation Labeling (Rangarajan, 2000; Berthod, 1996; Besag, 1986; Hummel and Zucker, 1993). In our experiments, our algorithm converges faster, while performing at least as accurately as the above mentioned methods. We show comparative results among BP, DA, DPA and ours.

For a more thorough analysis we generate synthetic energy functions. We pay more attention to the degree of connectedness and to the number of labels relative to the number of nodes. For non-planar graphs we generate their structure by picking an edge between any two sites with a certain probability p_{edge}. The first order potentials (as used in the product form of the Max-Product BP) are generated as uniform variables in the interval $[\epsilon, 1]$. Then, for each site we assume label number 1 to be the correct one, without loss of generality. Next we randomly group the sites into a number of disjoint sets (simulating the case when multiple objects or regions have to be classified simultaneously, a setup that relates to the discontinuity preserving energy functions from stereo). For pairs of connected sites (i, j), and the uniform random variable $p \in [\epsilon, 1]$, we set $Q_{ia;jb} = log(p/\epsilon)$ with probability p_0 and $Q_{ia;jb} = 0$ otherwise. If $a = b = 1$ and the pair of sites (i, j) are in the same set we have $p_0 = p_c$, otherwise we set $p_0 = p_w$ (for $p_c > p_w$). Thus we encourage second-order potentials between pairs of *correct* labels at sites in the same set to be on average larger than the rest of potentials. All experiments plotted on the same graph are generated using fixed parameters (e.g. number of nodes, labels per node, p_c, p_w, p_{edge}).

We ran experiments with different number of nodes and labels, different degree of connectedness and different values of p_c and p_w. Due to space limitation we only present sets of representative results (Figure (3)). Our algorithm, DA, and DPA use a parameter β that increases at each iteration ($\frac{1}{\beta}$ is known as the temperature in annealing approaches, which reaches 0 at the last iteration). For all three algorithms we use exactly the same β values at each iteration, such that $1/\beta$ decreases from 1 to 0.01 in equal steps. Also, for all algorithms we used the uniform x as the default initialization. Throughout the experiments our algorithm was always among the top performers along with DA, while converging much faster than both DA and DPA (Figure 5, 6, 7).

Belief Propagation is provably optimal for trees, but it has been applied successfully to graphs with loops. We compare our algorithm to Max Product BP and show that the performance of our algorithm is comparable with BP on trees, while consistently giving solutions of higher energies on highly connected graphs (Figures

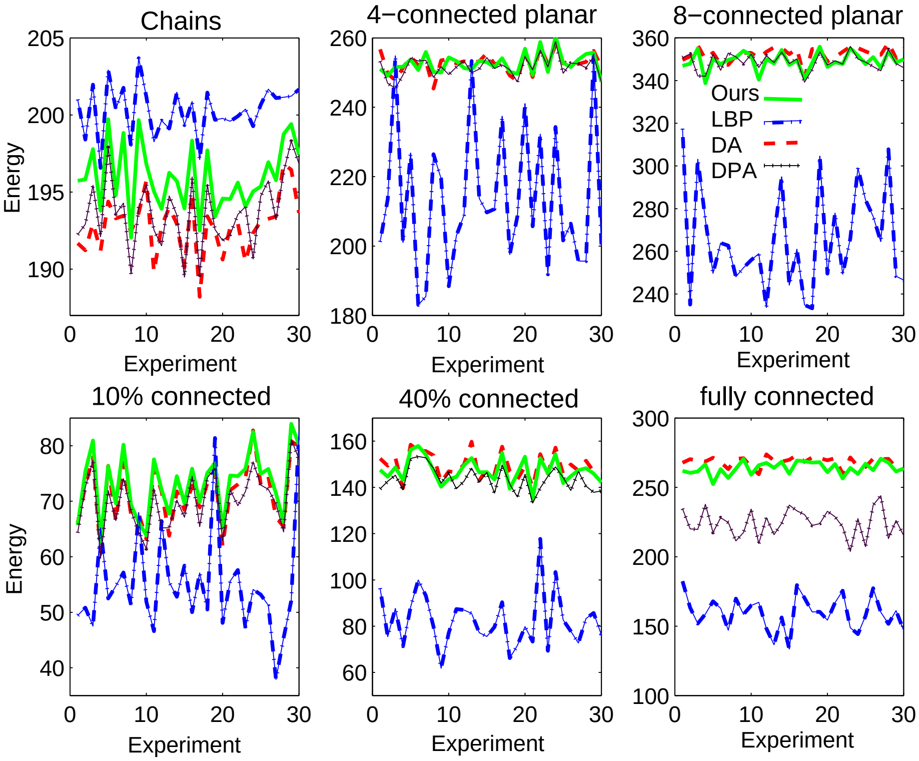

Figure 3. Results on 30 random experiments for different degree of graph connectedness. Top row: 100 nodes, 10 labels. Bottom row: 30 nodes, 30 labels. $p_c = 0.8$, $p_w = 0.4$, 180 iterations for each algorithm. Best viewed in color

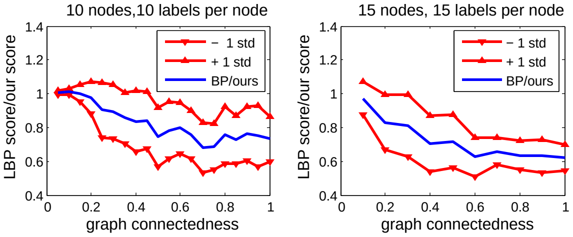

Figure 4. Mean and std values for E_{LBP}/E_{ours} for varying degree of connectedness, over 30 experiments.

Table 1. Results for fully connected graphs over 30 experiments (maximum of 30 iterations)

$nNodes$ $(= nLabels)$	avg of E_{BP}/E_{ours}	std of E_{BP}/E_{ours}
10	0.73	0.13
15	0.62	0.08
30	0.60	0.05

4 and 3). When the number of labels is comparable to the number of nodes, the performance of Max Product BP starts degrading significantly as we increase the degree of connectedness of the graphs. In Figure 4 we plot the mean ratio and standard deviation of BP energy vs. our energy over 30 experiments for different degrees of graph connectedness. The probability of edge generation ranges from 0.1 to 1 (fully connected). Both algorithms were allowed to run for a maximum of 30 iterations. We also observed that for a given degree of connectedness, Loopy BP's output energy degrades relative to ours, as we increase the number of nodes.

If it converges, Max Product Loopy BP is guaranteed to give a labeling with a larger energy than other assignments within a large neighborhood around that labeling (Weiss and Freeman, 2001). While this is an interesting theoretical result, it does not tell us anything about how often Loopy BP converges. In our experiments we actually found that Loopy BP rarely converges (Figures 5, 6, 7). We also noticed a very similar instability when experimenting with BP with sequential updates ($MPS-BP$) and momentum ($MPSM-BP$) for for dumping oscillations. The most stable version of BP was the Sum Product BP, with momentum and sequential updates ($SPSM - BP$) (Figure 8).

Our algorithm has an excellent average convergence performance, while seeming to be insensitive to the structure of the graph. DPA and DA also seem to

Table 2. Results for 8-connected planar graphs, 25 labels and 25 nodes over 30 experiments (maximum of 30 iterations)

BP *version*	*avg* of E_{BP}/E_{ours}	*std* of E_{BP}/E_{ours}
$MP-BP$	0.58	0.05
$MPS-BP$	0.71	0.09
$MPSM-BP$	0.93	0.08
$SPSM-BP$	0.75	0.05

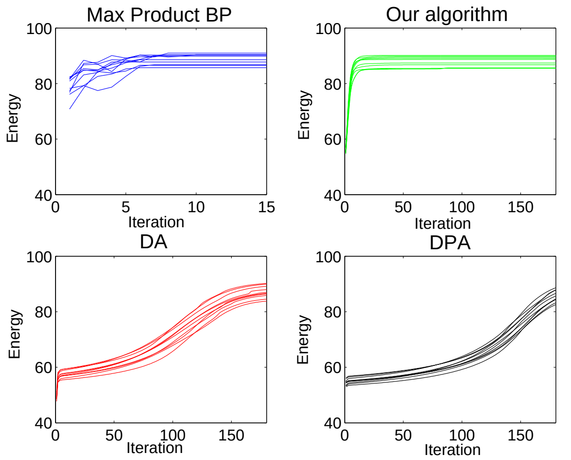

Figure 5. Experiments on chains, with 49 sites and 5 labels per site. The energy per iteration is plotted on the same 10 random experiments for each algorithm, with all parameters held constant

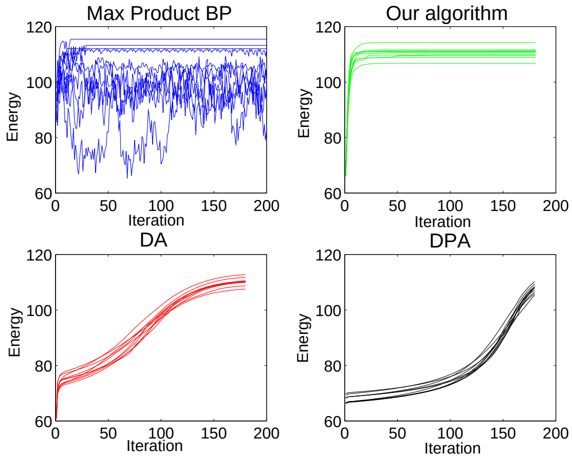

Figure 6. Experiments on planar graphs with 4-connected neighborhoods, with 49 sites and 5 labels per site. The energy per iteration is plotted on the same 10 random experiments for each algorithm, with all parameters held constant

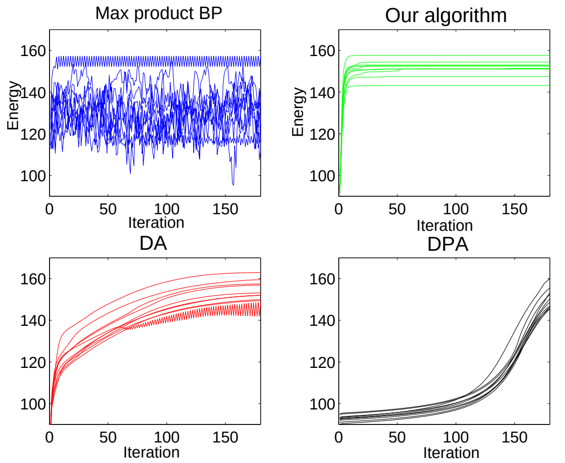

Figure 7. Experiments on planar graphs with 8-connected neighborhoods, with 49 sites and 5 labels per site. The energy per iteration is plotted on the same 10 random experiments for each algorithm, with all parameters held constant

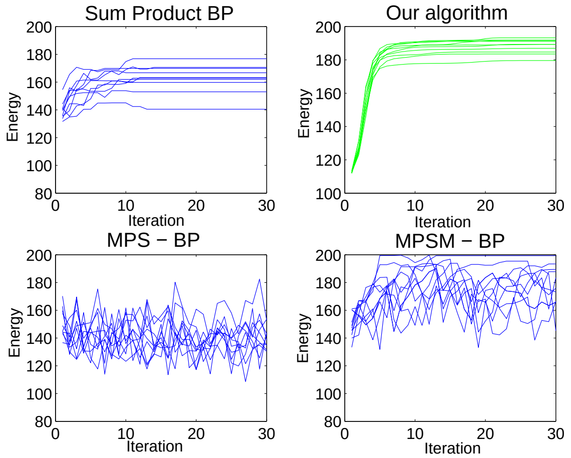

Figure 8. Experiments on fully connected graphs, 25 sites and 25 labels per site. The energy per iteration is plotted for the same 10 random experiments for each algorithm (all parameters held constant)

monotonically increase the energy after each iteration but their convergence is more sensitive to the graph structure. They maintained the same behavior even when the updates were done site-wise. We believe that the performance of our algorithm is due to the combination of its two steps, which have different roles that seem to complement each other. The role of the first step is to move the solution into the right direction with respect to the global optimum, as suggested by the optimality bounds. The next step improves the energy very rapidly at every iteration.

DA, DPA and our method require the same amount of memory, and roughly the same number of operations per iteration (= one updating of the vector x for all sites). In practice, our method takes fewer iterations to converge, which makes it more efficient. BP needs extra memory because it has to store about $nEdges * nLabels$ messages. Also, per iteration, BP needs to update all messages, while the other methods update only the $nNodes * nLabels$ beliefs in x. Our method took roughly the same number of iterations as BP to converge, which combined with its cheaper cost per iteration, makes it roughly $O(\frac{nEdges}{nNodes})$ times faster. For example, for a fully connected graph with 30 nodes and 30 labels per node, the $Matlab$ implementation of our algorithm took about 0.5 sec. on a laptop PC, while the BP implementation that we used (Kevin Murphy's Bayes Net Toolbox) took about 12 sec on the same problem.

8. Conclusions

We have presented an efficient approximate algorithm for energy optimization that has certain theoretical properties for arbitrarily structured graphs with arbitrary energy functions. Our experiments illustrate that our approach converges faster and it is less sensitive to the structure of the graphs than other existing methods, while being at least as accurate. For future work, it is worth investigating tighter theoretical bounds, that could explain better the high efficiency of our algorithm.

Acknowledgements

This work was performed in part under NSF Grant IIS0534962.The content does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

References

Baratchart, L., & Berthod, M., & Pottier, L. *Optimization of Positive Generalized Polynomials under lp Constraints* . Journal of Convex Analysis , 1998 Vol. 5, No. 2, pp. 353-379.

Baum L.E & Sell G. R, *Growth transformations for functions on manifolds* , Pacific Journal of Mathematics, Vol 27, No 2, pp. 361-363, 1967.

Berthod, M., & Kato Z., & Yu S., & Zerubia J. *Bayesian image classification using Markov random fields* . Image and Vision Computing. 14 285-295

Besag, J., *On the statistical analysis of dirty images*, Journal of Royal Statistics Society (1986)

Hummel, R.A, & S.W. Zucker, *On the foundations of relaxation labeling processes* , IEEE Trans. Patt. Anal. and Mach. Intell., Vol 5, No 3, pp 267-286

Kumar S., Hebert M., *Discriminative Random Fields: A Discriminative Framework for Contextual Interaction in Classification* , IEEE International Conference on Computer Vision Vol. 2, 2003, pp. 1150-1157

Meltzer T., Yanover C., Weiss Y., *Globally Optimal Solutions for Energy Minimization in Stereo Vision Using Reweighted Belief Propagation* , International Conference on Computer Vision, 2005

Murphy K. P, Weiss Y., Jordan M.I., *Loopy Belief Propagation for Approximate Inference: An Empirical Study* Uncertainty in Artificial Intelligence, 1999

Quattoni A., Collins M., Darrell T. *Conditional random fields for object recognition* , Advances in neural information processing systems, 2005

Rangarajan, A., *Self annealing and self annihilation: Unifying deterministic annealing and relaxation labeling* . Pattern Recognition, 2000

Wainwright M. J., Jaakkola T. S., Willsky A. S.*MAP estimation via agreement on (hyper)trees: Message-passing and linear-programming approaches*, IEEE Transactions on Information Theory, Vol. 51(11), pp 3697–3717. 2005

Yuille A.L., Rangarajan A., *The Concave-Convex Procedure (CCCP)* , Neural Computation, 2003

Weiss Y, Freeman W. T., *On the Optimality of Solutions of the Max-Product Belief Propagation Algorithm in Arbitrary Graphs* . IEEE Transactions on Information Theory, 2001