Revisiting MAP Estimation, Message Passing and Perfect Graphs

James R. Foulds **Nicholas Navaroli** **Padhraic Smyth** **Alexander Ihler**
Department of Computer Science
University of California, Irvine
{*jfoulds,nnavarol,smyth,ihler*}*@ics.uci.edu*

Abstract

Given a graphical model, one of the most useful queries is to find the most likely configuration of its variables. This task, known as the maximum a posteriori (MAP) problem, can be solved efficiently via message passing techniques when the graph is a tree, but is NP-hard for general graphs. Jebara (2009) shows that the MAP problem can be converted into the stable set problem, which can be solved in polynomial time for a broad class of graphs known as perfect graphs via a linear programming relaxation technique. This is a result of great theoretical interest. However, the article additionally claims that max-product linear programming (MPLP) message passing techniques of Globerson and Jaakkola (2007) are also guaranteed to solve these problems exactly and efficiently. We investigate this claim, show that it does not hold, and repair it with alternative message passing algorithms.

1 INTRODUCTION

Graphical models provide a compact representation of a probability distribution over a set of dependent random variables using a graph structure to represent dependence relationships among variables. A common task is the problem of finding a maximum a posteriori (MAP) configuration, i.e. an assignment of the variables that has the highest possible probability. The MAP problem is NP-hard in general (Shimony, 1994), but can be solved efficiently for certain classes of problems such as trees (Pearl, 1988) and maximum weight bipartite b-matching graphs (Bayati et al., 2005) using message passing algorithms such as max-product (Pearl, 1988). Such message passing algorithms exploit the independence structure of the graphical model by sending messages between nodes of the graph until convergence. When the graph contains cycles, max-product may not converge to the correct solution, or even converge at all, but frequently provides useful approximations. More recent variants of max-product (Globerson and Jaakkola, 2007; Sontag and Jaakkola, 2009) have convergence guarantees and may provide certificates of optimality, so that in many cases the solution can be guaranteed to be a MAP.

Jebara (2009) gives a sufficient condition and corresponding procedure for determining the exact MAP problem to be solvable in polynomial time. The procedure involves converting an MRF into an equivalent *NAND MRF*, which essentially encodes the MAP problem as a maximum weight stable set (independent set) problem, a well known discrete optimization task. If the graph of the NAND MRF is a *perfect* graph, the MAP can be found efficiently and exactly using a linear programming (LP) relaxation. Jebara (2009) thus expands the class of models known to have polynomial time MAP solutions. We provide some background on NAND MRFs and perfect graphs in Section 2, and summarize Jebara's main results in Section 3.

This paper is concerned with the message passing implications of Jebara (2009). Message passing algorithms for optimization are closely connected to linear programming relaxations, but may be more efficient than traditional LP solvers such as simplex or interior point methods since they are often able to exploit sparse problem structure in very large scale systems with many variables (Sontag et al., 2008; Wainwright et al., 2005) and can be easily parallelized. Jebara (2009) claims that the MPLP algorithm (Globerson and Jaakkola, 2007), a convergent variant of max-product related to linear programming, is also optimal on NAND MRFs with perfect graphs and further conjectures that the standard max-product algorithm may

Appearing in Proceedings of the 14^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2011, Fort Lauderdale, FL, USA. Volume 15 of JMLR: W&CP 15.

also be guaranteed to recover a MAP solution. We show that these claims do not hold, providing counterexamples that illustrate the issue in Section 4. We then repair the claims by applying more recent message passing algorithms and discuss their optimality guarantees in Section 5.1. We also propose a new message passing algorithm to directly optimize the dual of the linear program described in Section 5.2. We give experimental validation of our findings and compare various solutions in Section 6, and conclude with a discussion in Section 7.

2 BACKGROUND

We first describe several key background concepts, including perfect graphs and the NAND MRF construction of Jebara (2009). Their significance will become clear in Section 3.

2.1 Perfect Graphs

An undirected graph is considered to be a perfect graph if, for every induced subgraph, the clique number (size of the largest fully connected subset of nodes) is equal to its chromatic number (the minimum number of colors needed so that every node is labeled with a color and no two adjacent nodes share the same color) (Berge, 1963). Berge formulated the strong perfect graph conjecture, stating that a graph is perfect if and only if it is a Berge graph: a graph that does not contain an odd hole (chordless cycle of odd length 5 or greater), and whose complement contains no odd hole. Thus, a graph is a Berge graph if and only if its complement is also a Berge graph. The strong perfect graph conjecture was proven over 40 years later by Chudnovsky et al. (2006), and is now known as the strong perfect graph theorem. Note that this theorem implies the earlier proven "weak" perfect graph theorem (Lovász, 1972), namely that a graph is perfect if and only if its complement is perfect. Chudnovsky et al. (2005) created an algorithm to check if any graph is a Berge graph (and therefore a perfect graph), which has time complexity $O(n^9)$, where n is the number of vertices.

2.2 NAND Markov Random Fields

A Markov random field (MRF) connects a probability distribution over a collection of variables $X = \{x_1, \ldots, x_n\}$ to an undirected graph G. Each variable x_i is associated with a vertex v_i of G. The graph is said to be consistent with a probability distribution if and only if it factors into a product of functions, called potential functions, defined only over the cliques C of the graph, so that

$$P(X) = \frac{1}{Z} \prod_{c \in C} \psi_c(X_c) \quad \text{where } X_c = \{x_i : v_i \in c\}$$

and Z is the partition function, chosen to normalize $P(X)$. The structure of a MRF does not uniquely determine a factorization, and so it is sometimes useful to use factor graphs (Kschischang et al., 2001), another type of graphical model, to more explicitly represent the factorization. In factor graphs, each potential function ψ is explicitly represented in the graph as a "factor node" (drawn in our figures as a square) which is connected to the variable nodes associated with its arguments.

A NAND Markov random field (NMRF) (Jebara, 2009) is a special kind of MRF in which

- Each variable x_i is binary, $x_i \in \{0, 1\}$.
- Potential functions $\psi_i(x_i)$ are associated with variable nodes and represented with a nonnegative value f_i.
- Potential functions $\psi_{ij}(x_i, x_j)$ are associated with the edges of the graph which are pairwise nand gates.

$$\psi_i(x_i) = \begin{cases} 1 & x_i = 0 \\ \exp(f_i) & x_i = 1 \end{cases}$$

$$\psi_{ij}(x_i, x_j) = \begin{cases} 0 & x_i = x_j = 1 \\ 1 & \text{otherwise} \end{cases}$$

The MAP problem for an NMRF is exactly the maximum weight stable set problem for its graph, with the weight of node i equal to f_i. In this formulation, the task is to find the set of nodes with the maximum total weight such that the nodes form a stable set, i.e., there are no two adjacent nodes in the set. NMRFs can be easily used to encode matching problems, including the marriage problem and its generalizations. However, Jebara (2009) shows their more general applicability by providing a procedure for converting any arbitrary MRF with graph G into an equivalent NMRF with graph $\mathcal{G}$. In the procedure, a node of $\mathcal{G}$ is created for every configuration of variables in each potential function of G's MRF. Nodes that correspond to conflicting variable assignments are then connected via nand edges. Each NMRF node weight is the value of the corresponding factor and configuration in the original graph. The original factor can be rescaled to ensure that f_i is nonnegative.

3 SOLVING MAP USING NAND MRFS AND PERFECT GRAPHS

The primary result of Jebara (2009) is a procedure for finding the MAP in polynomial time, for certain classes of graphs. The procedure is to first convert a graphical model with graph G into an equivalent NAND MRF with graph $\mathcal{G}$, check whether $\mathcal{G}$ is perfect, and if so construct a linear programming problem which is guaranteed to produce the MAP solution in polynomial time. All steps of the procedure can be performed in polynomial time, and thus the procedure is also polynomial time.

The NAND MRF is used to construct a program of the form

$$x^* = \arg\max_x f^T x \tag{1}$$
$$\text{s.t. } Ax \le \mathbf{1} \qquad \text{and} \qquad x \ge \mathbf{0} \ .$$

where $x = [x_1, \ldots, x_n]^T$ is the vector of values for the variables in $\mathcal{G}$, $f = [f_1, \ldots, f_N]^T$ are the weights for each node (i.e., $f_i = \log \psi_i(x_i = 1)$), and $\mathbf{1}$ and $\mathbf{0}$ are all-ones and all-zeros vectors. The matrix A is the maximal clique incidence matrix: a value of 1 at row c and column n indicates that node n on the graph is a part of maximal clique c, and 0 otherwise. Grötschel et al. (1988) refer to the constraint set of this program as the *clique constrained stable set polytope.* We therefore refer to the program (1) as the *clique constrained LP* (CC_LP). If the x_i are integer, $x_i \in \{0, 1\}$, then $f^T x$ equals the log-likelihood of configuration x and the constraint matrix A guarantees that only one node in each maximal clique is assigned a value of 1, enforcing the NAND property of the NMRF. Solving CC_LP as an integer program therefore finds the MAP configuration of the NMRF. Relaxing CC_LP to be a linear program allows it to be solved in polynomial time (Grötschel et al., 1988). If an integral solution is recovered, this must be the MAP configuration. Interestingly, CC_LP has an integral solution if and only if the graph is perfect (Chvátal, 1975).

The computational challenge in formulating CC_LP is finding the maximal cliques for a given NMRF. This is NP-hard in general, but can be done in polynomial time for perfect graphs (Grötschel et al., 1988).

Jebara (2009) also makes a claim related to message passing algorithms on NAND MRFs: that convergent variants of max-product, in particular the max-product linear program (MPLP) algorithm of Globerson and Jaakkola (2007), is also guaranteed to find the MAP on an NMRF. The primary contribution of this paper is to show that the MPLP algorithm does not have such guarantees, and attempt to repair the claim by describing alternative message passing algorithms that are guaranteed to find the MAP on NMRFs whose graph is perfect. First, we briefly describe MPLP and its usage in Jebara (2009).

The MPLP algorithms of Globerson and Jaakkola (2007) are based on a linear programming relaxation of the MAP problem. For a given MRF with graph G and potential functions defined on the edges of the graph $\theta_{ij}(x_i, x_j)$, Globerson and Jaakkola (2007)'s LP formulation, known as MAP Linear Programming Relaxation (MAPLPR), is the LP relaxation of the following integer program:

$$\mu^* = \arg\max_{\mu \in \mathcal{M}_L(G);\ \mu \text{ integral}} \mu^T \theta \ , \tag{2}$$

where μ is a vector of indicator functions for each possible assignment to each edge and each node in the graph, the elements of θ are chosen such that $\mu^T \theta$ equals the log-likelihood of configuration μ if μ is integral, and $\mathcal{M}_L(G)$ is the so-called *local marginal polytope*, which ensures that the μ satisfy the simple local consistency relationships of marginalization: for all $(i, j) \in E$ and all values x_i, x_j

$$\mu_{ij}(x_i, x_j) \ge 0, \qquad \sum_{\hat{x}_i} \mu_{ij}(\hat{x}_i, x_j) = \mu_j(x_j),$$
$$\sum_{\hat{x}_j} \mu_{ij}(x_i, \hat{x}_j) = \mu_i(x_i), \qquad \sum_{x_i} \mu_i(x_i) = 1.$$

For NMRFs, the θ vector is given by $\theta_i(x_i) \equiv 0$ and pairwise $\theta_{ij}(x_i, x_j)$ as

$$\theta_{ij}(x_i, x_j) = \begin{cases} 0 & \text{if } (x_i, x_j) = (0, 0) \\ \frac{f_j}{|\text{Ne}(j)|} & \text{if } (x_i, x_j) = (0, 1) \\ \frac{f_i}{|\text{Ne}(i)|} & \text{if } (x_i, x_j) = (1, 0) \\ -\infty & \text{if } (x_i, x_j) = (1, 1) \end{cases} \tag{3}$$

where Ne(i) are the neighbors of node i.

MPLP comes in two forms, one which iterates over edges of the graph (EMPLP) and one which iterates over nodes (NMPLP) when updating messages. Both algorithms are equivalent to optimizing the dual problem of MAPLPR via coordinate descent, and therefore monotonically decrease the dual objective function. Consequently they are guaranteed to converge to a fixed point for arbitrary graphs. MPLP can also be generalized to iterate between cliques and nodes in the factor graph representation.

4 MPLP IS NOT OPTIMAL FOR NMRFS

The argument for the claim regarding convergent message passing (Jebara, 2009) is roughly as follows: (1) the CC_LP formulation is integral for perfect graphs, so linear programming relaxation techniques will still find the MAP; (2) the MAPLPR formulation of the LP is equivalent to CC_LP, so any technique that solves the MAPLPR for a perfect graph will find the MAP;

and (3) MPLP is such a technique, and so MPLP recovers the MAP.

There are several issues with this argument. The least serious of these involves part (1). The theorem of Chvátal (1975) shows that CC_LP has at least one integral solution when the graph is perfect, but does this not preclude the existence of additional non-integral solutions that could cause LP relaxation techniques to fail. The following theorem shows that we must also require the MAP to be unique to guarantee that linear programming relaxation techniques will find the MAP.

Theorem 4.1. *The MAP is unique iff no non-integral solutions to CC_LP exist, for NMRFs with perfect graph G.*

Sketch of proof. The feasible region of CC_LP is the interior of the convex hull of the incidence vectors of the stable sets of perfect graph G (Chvátal, 1975). It can be shown that for primal feasible x, the CC_LP objective function for x is the weighted average of the objective function for the incidence vectors of the stable sets, with the weights corresponding to the weights for the convex combination used to generate x. The set of optimal solutions is therefore exactly the set of convex combinations of the MAP configurations. □

A more critical issue, however, arises in (2) of the argument. CC_LP and MAPLPR are equivalent integer programs, where CC_LP uses constraints on the cliques, and MAPLPR uses constraints on pairs of adjacent nodes. However, their LP relaxations are not equivalent. Crucially, the integrality of CC_LP does not imply the integrality of MAPLPR. We demonstrate this with a small example.

Consider Figure 2. This graph is the complement of a bipartite graph, and is therefore perfect. The values of f_{x_i} below the graph are the logs of the potential functions for each node. The values for all $\mu_{ij}(x_i, x_j)$ are shown, along with intermediate products for computing the MAPLPR objective function $\mu^T\theta$. The values for θ are computed using Equation (3). The exact MAP, which CC_LP correctly recovers, is the assignment $[A, B, C, D, E] = [0, 1, 0, 1, 0]$. The log likelihood of this configuration is $f_B + f_D = 8$. The μ assignment given is an optimal solution to MAPLPR with non-integral values and an objective function of $\mu^T\theta = 9$, greater than the log likelihood for the exact MAP. The non-integrality of this solution is seen in nodes C, D, E, all of which were assigned weights of 0.5.

The reason that MAPLPR is not necessarily integral is shown by Figure 1. Although CC_LP and MAPLPR are equivalent integer programs, the pairwise constraints of MAPLPR allow additional non-integral vertices that are excluded in the clique constraints of CC_LP. It is known in the combinatorial optimization literature that clique constraints are sufficient for integrality for the stable set problem (i.e. MAP for NMRFs) if and only if G is a perfect graph (Grötschel et al., 1988). A special case is that of bipartite graphs, whose maximal cliques are pairs of nodes; it is known that pairwise constraints are sufficient if and only if G is bipartite.

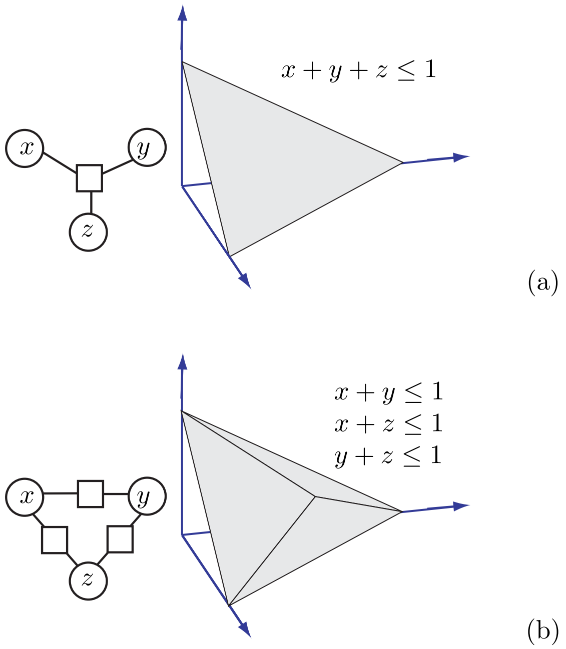
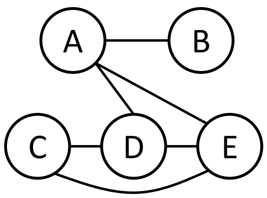


Figure 1: (a) Joint constraints encoding a three-variable NAND relationship, and (b) pairwise version of the same constraints. Although both have the same set of integer constraints, their linear relaxations are not equivalent; the pairwise version introduces additional non-integral vertices.

5 ALGORITHMS

We would like to repair the claim of Jebara (2009) by finding a message passing algorithm that is guaranteed to find the MAP for NMRFs with perfect graphs. We show that a higher-order dual decomposition method with subgradient updates, referred to here as DD-SG, is such an algorithm. Additionally, we introduce a new convergent message passing algorithm, CD2MP, based on coordinate descent on pairs of cliques in the dual of CC_LP. Unlike DD-SG, CD2MP is monotonic in the dual objective function. We give some restricted optimality guarantees for this algorithm, and conjecture that its fixed points must recover the MAP for NMRFs with perfect graphs.



$f_A = 2,$
$f_B = f_D = 4,$
$f_C = f_E = 3$

$$
\begin{aligned}
\theta_{AB} &= \begin{bmatrix} 0 & 4/1 \\ 2/3 & -\infty \end{bmatrix} & \mu_{AB} &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} & &\Rightarrow \tfrac{4}{1} \times 1 \\
\theta_{AD} &= \begin{bmatrix} 0 & 4/3 \\ 2/3 & -\infty \end{bmatrix} & \mu_{AD} &= \begin{bmatrix} .5 & .5 \\ 0 & 0 \end{bmatrix} & &\Rightarrow \tfrac{4}{3} \times \tfrac{1}{2} \\
\theta_{AE} &= \begin{bmatrix} 0 & 3/3 \\ 2/3 & -\infty \end{bmatrix} & \mu_{AE} &= \begin{bmatrix} .5 & .5 \\ 0 & 0 \end{bmatrix} & &\Rightarrow \tfrac{3}{3} \times \tfrac{1}{2} \\
\theta_{CD} &= \begin{bmatrix} 0 & 4/3 \\ 3/2 & -\infty \end{bmatrix} & \mu_{CD} &= \begin{bmatrix} 0 & .5 \\ .5 & 0 \end{bmatrix} & &\Rightarrow \tfrac{4}{3} \times \tfrac{1}{2} + \tfrac{3}{2} \times \tfrac{1}{2} \\
\theta_{CE} &= \begin{bmatrix} 0 & 3/3 \\ 3/2 & -\infty \end{bmatrix} & \mu_{CE} &= \begin{bmatrix} 0 & .5 \\ .5 & 0 \end{bmatrix} & &\Rightarrow \tfrac{3}{3} \times \tfrac{1}{2} + \tfrac{3}{2} \times \tfrac{1}{2} \\
\theta_{DE} &= \begin{bmatrix} 0 & 3/3 \\ 4/3 & -\infty \end{bmatrix} & \mu_{DE} &= \begin{bmatrix} 0 & .5 \\ .5 & 0 \end{bmatrix} & &\Rightarrow \tfrac{3}{3} \times \tfrac{1}{2} + \tfrac{4}{3} \times \tfrac{1}{2}
\end{aligned}
$$

Figure 2: A counterexample showing that MAPLPR can be non-integral for NMRFs with perfect graphs. The integer solution is $B = D = 1$, $A = C = E = 0$ with value 8; MAPLPR finds a fractional solution with value 9.

5.1 Higher-Order Message Passing

One way to repair the non-equivalence of pairwise MPLP is to operate on the maximal clique graph, in which each maximal clique of the NMRF corresponds to a factor representing a joint NAND constraint. Analagous to Figure 2, we define factors over d variables by 2^d entries. Assignments violating the NAND constraint have value $-\infty$, the assignment where all variables are zero has value zero, and assignments where only variable i is one have value f_i/N_i, where N_i is the number of maximal cliques in which i participates.

A generalization of MPLP message passing to higher-order clique updates (referred to as GMPLP) is discussed briefly in Globerson and Jaakkola (2007); however, it is perhaps easier to use the equivalent LP associated with the tree-reweighted max-product (TRW) algorithm (Wainwright et al., 2005) and sequential, monotone update sequence of Kolmogorov (2006). Sontag and Jaakkola (2009) discuss the equivalence of the LP relaxation dual forms of TRW, MPLP, and several other algorithms. These methods can be understood within the dual decomposition framework, in which the dual formulation of the optimization problem can be decomposed into smaller subproblems that are tied together via Langrange multipliers.

We consider a TRW-like dual decomposition method using fixed point updates (DD-FP), that operates by creating a collection of trees which span the full graph, and associating with each tree t a set of parameters θ^t such that they sum to the original model, $\sum_t \theta^t = \theta$. Each tree is solved separately, providing an upper bound on the MAP, and the algorithm then minimizes this upper bound over the allocation of θ to each tree. DD-FP provides a monotone update by sequentially visiting each node and edge and "merging" its copies. We choose the individual factors as our collection of trees; with our previous definition, we need only add local parameters $\theta_i^c(0), \theta_i^c(1)$ for each clique c and variable i, with $\sum \theta_i^c(j) = 0$ for all i, j. To update, we compute the max-marginals at each node by

$$\gamma_{ci}(1) = \theta_i^c(1) + \sum_{j \neq i} \theta_j^c(0)$$

$$\gamma_{ci}(0) = \max \left[\sum_j \theta_j^c(0), \max_{j \neq i} \left[\theta_j^c + \sum_{k \neq j} \theta_k^c \right] \right]$$

and update the local parameters by

$$\theta_i^c(k) = \theta_i^c(k) - \gamma_{ci}(k) + \frac{1}{N_i} \sum_{c'} \gamma_{c'i}(k)$$

This efficient update, linear in clique size, is a consequence of the NAND form of the factors. It can be shown that optimizing over the θ^c's corresponds to solving the dual of CC_LP, and that DD-FP corresponds to a coordinate descent algorithm on the dual (Kolmogorov, 2006).

Unfortunately, while this fixes the issue of enforcing pairwise versus clique constraints and while DD-FP (or equivalently, GMPLP) is monotone and thus convergent, higher-order fixed point algorithms break part (3) of the argument, that convergence implies optimality. This argument used results showing that binary, pairwise graphs result in partially decodable fixed points (Kolmogorov and Wainwright, 2005). In more general systems however, such coordinate descent algorithms can suffer from fixed points (called *weak tree agreement* points) that are not optima of the corresponding LP (Kolmogorov, 2006).

Another solution is to use dual decomposition methods with subgradient updates (DD-SG) (e.g. Komodakis and Paragios (2008)). Rather than defining a fixed point procedure, these methods solve the individual subproblems (finding an optimal configuration of each tree) then update the parameters using the solution, which corresponds to a subgradient update of the dual parameters θ^t. Since these updates correspond to gradient descent, given an appropriate sequence of step

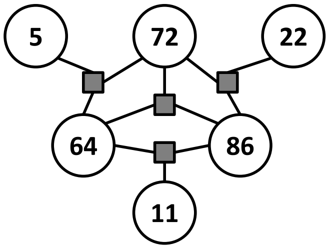


Figure 3: A perfect graph, represented as a factor graph, in which fixed point methods fail to recover the MAP. Node weights are given inside each node. The log likelihood of the MAP is 5 + 86 = 91.

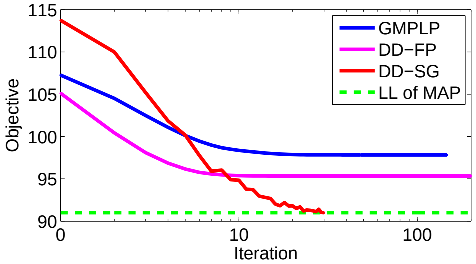


Figure 4: Suboptimality of fixed point methods. DD-SG's dual objective converges to the optimal value and recovers the exact MAP. GMPLP and DD-FP's dual objectives converge to sub-optimal fixed points.

sizes they can be guaranteed to optimize the dual of CC_LP and thus find the MAP. DD-SG can still be thought of as message-passing, since it requires only local operations (solving each factor individually and updating the local θ^c).

Figure 3 shows a perfect graph in which fixed point methods do not recover the exact MAP. The behavior of the fixed point and subgradient solvers on this graph is displayed in Figure 4. DD-FP and GMPLP become stuck at weak tree agreement fixed points and do not return the MAP, while DD-SG finds the correct solution. Because fixed point solvers often converge faster than subgradient methods, for this and subsequent experiments we used interleaved DD-FP and DD-SG steps to improve performance while keeping the guarantee of optimality.

5.2 CD2MP

Although DD-SG is optimal for NMRFs with perfect graphs, we would like to have a monotonic message passing algorithm with this property. We derive a message passing algorithm that solves the CC_LP via block coordinate descent in the dual with updates on *pairs* of overlapping cliques and prove some optimality properties for its fixed points in Section 5.2.1. The CC_LP is a *packing* LP, so its dual is a *covering* LP:

$$\begin{aligned} y^* &= \arg\min \sum y \\ \text{s.t. } & A^\intercal y \geq f \ , \ y \geq 0 \ . \end{aligned} \tag{4}$$

where the dimensionality of y is the number of maximal cliques. Let c_a, c_b be overlapping cliques. We want to perform the pairwise coordinate descent update for $y_s = \{y_a, y_b\}$, fixing $y \setminus y_s$. Let

$$h_i = max(0, (f_i - \sum_{\{j|i \in c_j, y_j \notin y_s\}} y_j)) \forall i \in \cup y_s$$

$$s_a = \max_{i \in c_a \setminus c_b} h_i \ , \ s_b = \max_{i \in c_b \setminus c_a} h_i \ , \ q = \max_{i \in c_b \cap c_a} h_i$$

The relevant part of the LP is:

$$\begin{aligned} y_s^* &= \arg\min_{y_a, y_b} y_a + y_b \\ \text{s.t. } & y_a \geq s_a \ , \ y_b \geq s_b \ , \ y_a + y_b \geq q \ . \end{aligned} \tag{5}$$

Clearly $y_a^* + y_b^* = \max\{s_a + s_b, q\}$. This can be obtained via the update rule

$$y_a \leftarrow \max\{s_a, 0.5(s_a - s_b + q)\} \ , \tag{6}$$

and similarly for y_b with a and b reversed. This update is repeated for each pair of overlapping cliques until convergence. Note that convergence is guaranteed because the algorithm monotonically decreases the dual objective function. We will refer to the algorithm as CD2MP, short for *pairwise coordinate descent message passing.*

The CD2MP algorithm has a message passing interpretation, in which messages are communicated between pairs of cliques, and between those cliques and the nodes that they contain. The messages, and their semantics in terms of the covering interpretation of the CC_LP are described in Table 1. In terms of these messages, we can rewrite Equation 6 as

$$y_a \leftarrow \max\{\lambda_{ab}, 0.5(\lambda_{ab} - \lambda_{ba} + \max_{i \in c_a \cap c_b} \lambda_{iab})\} \ . \tag{7}$$

The CD2MP algorithm is a useful tool for solving the dual problem. However, we set out to solve the primal problem, so a solution is only useful if we can recover the primal solution. Suppose y^* is an optimal solution to the dual. By the complementary slackness theorem, all x_i whose corresponding constraints in the dual are loose for y^* (their slack variables are greater than zero) must be zero in the primal optimal solution. Any x_i

Sender	**Receiver**	**Message**	**Covering LP Interpretation**
Clique a	Node i	$\lambda_{ai} = y_a$	I have λ_{ai} units of a available.
Node i	Clique a	$\lambda_{ia} = \max\{0, f_i - \lambda_i^{-\{a\}}\}$	I need $\geq \lambda_{ia}$ units of a.
Node i	Cliques a, b	$\lambda_{iab} = \max\{0, f_i - \lambda_i^{-\{a,b\}}\}$	I need $\geq \lambda_{iab}$ units between a and b.
Clique a	Clique b	$\lambda_{ab} = \max_{i \in c_a \setminus c_b} \lambda_{ia}$	I must have $\geq \lambda_{ab}$ units of a

Table 1: Message Passing Interpretation of CD2MP. In the above, $\lambda_i^{-s} = \sum_{d:i\in c_d, d\notin s} \lambda_{di}$, the sum of the messages from cliques that i is in, except the set of cliques s.

whose corresponding constraint is tight and is not adjacent to any other x_i with a tight constraint must have value one, since this gives a higher value for the decoded solution and does not violate the constraints. If there are sets of x_i with tight constraints whose induced subgraph is connected, decoding is non-trivial for this part of the solution. However we obtain partial decoding in the sense that the other decoded x_is must belong to an optimal solution.

5.2.1 Properties of CD2MP

A key property of CD2MP is that it monotonically improves the dual objective function, so it is guaranteed to converge to a fixed point. We can prove the following properties of these fixed points:

Theorem 5.1. *Fixed Points of CD2MP are Optimal with Unambiguous Decoding*

Sketch of Proof. Let y be a converged solution that is uniquely decodable to integral primal solution x (i.e. there are no two adjacent nodes with tight constraints). We want to show that y and x are optimal. By duality, it is sufficient to show that they have the same objective function value. The dual solution has value $\sum y$. The decoded primal solution has value $\sum_{i:x_i=1} f_i = \sum_{j:\exists i: i\in c_j, \sum_{k:i\in c_k} y_k = f_i} y_j$, the sum of the y_i that contain nodes with tight constraints. It remains to show that the cliques c_i that contain no nodes with tight constraints have $y_i = 0$. Suppose clique c_a has no nodes with tight constraints, and c_b overlaps with c_a. Since y is a fixed point, either $y_a + y_b = \max_{i\in c_a \cap c_b} \lambda_{iab}$ or $y_a = \lambda_{ab}$, $y_b = \lambda_{ba}$. In either case it can be shown that $y_a = 0$. □

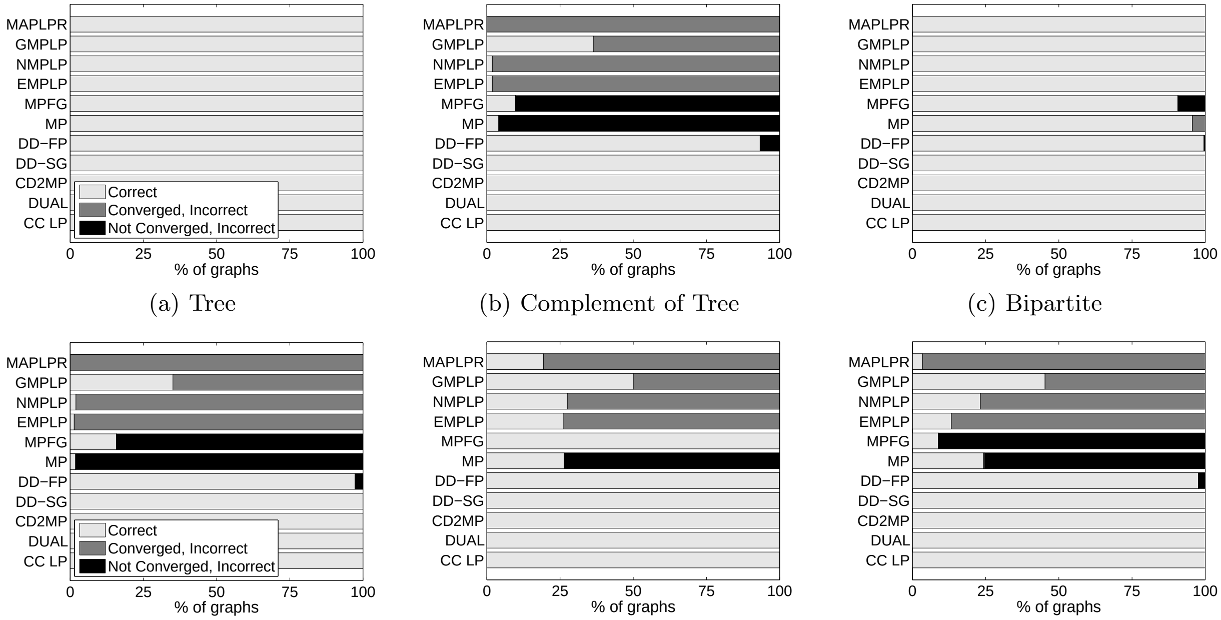
Theorem 5.2. *Fixed points of CD2MP are dual optimal for perfect graphs with clique number equals two.*

Sketch of Proof. Assume that CD2MP has converged to y for a problem with perfect graph G. Suppose that S is a maximal set of nodes in G that have tight constraints and form a connected graph. Let y_s be the set of maximal cliques containing nodes in S. From duality and the previous theorem, it is sufficient to show that for any such S there exists a legal decoding x s.t. $\sum_{i\in S, x_i=1} f_i = \sum_{y\in y_s} y$. For all nodes i in S, $f_i = \sum_{i\in c_j} y_j$, so it is enough to show that it is possible to enable exactly one node (whose constraint is tight) per maximal clique in y_s, which would give us $\sum_{i\in S, x_i=1} f_i = \sum_{x_i=1}\sum_{i\in c_j} y_j = \sum_{y\in y_s} y$. Since G is perfect, the subgraph induced by S is also perfect. As the clique number equals two and the subgraph induced by S is perfect, its color number also equals two. The color with the largest number of nodes is the maximum stable set, and must cover every clique, and thus every maximal clique (since the maximal cliques are just the cliques in this case). □

Both of these properies are shared by DD-FP and GMPLP, which can also be interpreted as coordinate descent in the dual; however, in the experimental section we find that CD2MP is successful even when DD-FP and GMPLP are not. The primary difference between the algorithms is that the coordinate space of the CD2MP updates is larger than that of GMPLP. GMPLP updates messages from a single clique to its variables, while DD-FP updates the local parameters of a single variable in its neighboring cliques; both can be regarded as individual updates of single-variable parameters. In contrast, CD2MP acts on *pairs* of cliques, giving it a strictly larger set of update directions that include *jointly* changing the parameters of all variables in the overlap of the two cliques. We conjecture that the fixed points of the CD2MP algorithm are dual optimal for perfect graphs in general. This conjecture is supported by the experimental results that we present in the next section.

6 EXPERIMENTS

The optimality and convergence properties of the algorithms described in this paper were investigated using the classes of perfect graphs used in Jebara (2009): tree, complement of tree, bipartite, complement of bipartite, line graph of bipartite, and complement of line graph of bipartite. For each type of perfect graph, 1000 graphs containing between 8 and 50 nodes were generated. Each node was assigned a weight chosen uniformly at random, $f_n \sim \mathrm{U}[0, 1]$. Trees were formed by repeatedly joining connected components at nodes chosen uniformly at random. Bipartite graphs were formed by creating two sets of nodes; for each node in the first set, a random number of connections to the



(a) Tree

(b) Complement of Tree

(c) Bipartite

(d) Complement of Bipartite

(e) Line Graph of Bipartite

(f) Complement, Line Graph of Bipartite

Figure 5: Outcome of applying each algorithm to each type of graph. Outcomes are (1) recovered exact MAP, (2) recovered sub-optimal solution, and (3) did not converge and recovered sub-optimal solution.

second set were added. Convergence was determined by comparing the change in the objective between two iterations to a threshold. If an algorithm did not converge after 5000 iterations, it was terminated and the solution returned by the last iteration was used.

Figure 5 shows the results of applying each algorithm to each graph. As expected, CC_LP and its dual recovered the MAP for each graph via an interior-point method. Because interior-point methods are not message passing algorithms, having loops in the perfect graph has no effect. DD-SG also recovered the MAP for each graph. CD2MP always recovered the MAP, supporting our conjecture regarding its optimality for finding the MAP for NMRFs with perfect graphs.

DD-FP and GMPLP failed to recover the MAP in some cases, although they find the MAP for the majority of generated graphs. EMPLP, NMPLP, and MAPLPR always recovered the MAP for trees and bipartites; optimality is only guaranteed for these graphs in which pair-wise constraints correspond to maximal clique constraints. Max-product (MP) only always recovered the MAP for trees, the only type of perfect graph tested that does not contain loops. Max-product over the factor graphs (MPFG) behaved similarly.

7 DISCUSSION

There is considerable interest in the theoretical properties of techniques for solving queries on graphical models. A question of particular interest is to identify the set of problems that message passing algorithms can solve optimally, or can be solved efficiently by other techniques such as LP relaxations. Jebara (2009) shows that MRFs with pairwise nand clique functions (NMRFs) are amenable to polynomial time MAP recovery by LP relaxation methods for a special class of graphs called perfect graphs, and gives a method for converting an arbitrary MRF into an equivalent NMRF. We applied dual decomposition to NMRFs with perfect graphs, showing that DD-SG is guaranteed to find the MAP. Thus, our contribution repairs the claim of Jebara (2009) that there exists a message passing algorithm that can recover the MAP for MRFs whose equivalent NMRFs are perfect. We also introduced a new convergent message passing algorithm for NMRFs, CD2MP, and proved some properties regarding the optimality of its fixed points. Experimental results on randomly generated perfect graphs support the conjecture that CD2MP always recovers the MAP for NMRFs with perfect graphs. Future work is to prove or disprove this conjecture.

Acknowledgements

This work was supported in part by ONR/MURI under grant number N00014-08-1-1015 (JF, NN, PS).

References

M. Bayati, D. Shah, and M. Sharma. Maximum weight matching via max-product belief propagation. *IEEE International Symposium on Information Theory*, 2005.

C. Berge. *Six Papers on Graph Theory*, chapter Perfect graphs, pages 1–21. Calcutta: Indian Statistical Institute, 1963.

M. Chudnovsky, G. Cornuéjols, X. Liu, P. Seymour, and K. Vušković. Recognizing berge graphs. *Combinatorica*, 25(2):143–186, 2005.

M. Chudnovsky, N. Robertson, P. Seymour, and R. Thomas. The strong perfect graph theorem. *Annals of Mathematics-Second Series*, 164(1):51–230, 2006.

V. Chvátal. On certain polytopes associated with graphs. *J. Combin. Theory Ser. B*, 18:138–154, 1975.

A. Globerson and T. Jaakkola. Fixing max-product: Convergent message passing algorithms for MAP LP-relaxations. In *Advances in Neural Information Processing Systems*, volume 21, 2007.

M. Grötschel, L. Lovász, and A. Schrijver. *Geometric algorithms and combinatorial optimization.* Algorithms and Combinatorics. Springer–Verlag, 1988.

T. Jebara. Map estimation, message passing, and perfect graphs. In *Uncertainty in Artificial Intelligence*, 2009.

V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(10):1568–1583, 2006. ISSN 0162-8828.

V. Kolmogorov and M. Wainwright. On the optimality of tree-reweighted max-product message-passing. In *Uncertainty in Artificial Intelligence*, pages 316–323, 2005.

N. Komodakis and N. Paragios. Beyond loose lp-relaxations: Optimizing mrfs by repairing cycles. In *ECCV '08: Proceedings of the 10th European Conference on Computer Vision*, pages 806–820, Berlin, Heidelberg, 2008. Springer-Verlag. ISBN 978-3-540-88689-1.

F. Kschischang, B. Frey, and H. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 47:498–519, 2001.

L. Lovász. Normal hypergraphs and the weak perfect graph conjecture. *Discrete Mathematics*, 2:253–267, 1972.

J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference.* Morgan Kaufmann San Mateo, CA, 1988.

S. Shimony. Finding MAPs for belief networks is NP-hard. *Artificial Intelligence*, 68(2):399–410, 1994.

D. Sontag and T. Jaakkola. Tree block coordinate descent for MAP in graphical models. In *Proceedings of the 12th AISTATS*, 2009.

D. Sontag, T. Meltzer, A. Globerson, T. Jaakkola, and Y. Weiss. Tightening lp relaxations for map using message passing. In *Uncertainty in Artificial Intelligence*, pages 503–510, 2008.

M. Wainwright, T. Jaakkola, and A. Willsky. Map estimation via agreement on trees: message-passing and linear programming. *IEEE Transactions on Information Theory*, 51(11):3697–3717, 2005.