A Combined LP and QP Relaxation for MAP

Patrick Pletscher
ETH Zurich, Switzerland
pletscher@inf.ethz.ch

Sharon Wulff
ETH Zurich, Switzerland
sharon.wulff@inf.ethz.ch

Abstract

MAP inference for general energy functions remains a challenging problem. Linear programming (LP) relaxations for MAP incorporate pairwise auxiliary variables encoding assignments for edges. We introduce LPQP, a MAP formulation which includes a penalty on the Kullback-Leibler divergence between the auxiliary variables and the corresponding quadratic terms formed by unary variables. An efficient DC algorithm is derived for minimizing the resulting non-convex formulation. The core task of the algorithm reduces to a variant of the norm-product belief propagation with modified unary potentials. Experiments on synthetic and real-world data show substantial improvements over the standard LP relaxation.

1 Introduction

In this work we study a discrete pairwise energy minimization problem for a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ of the form

$$\min_{\boldsymbol{x}} \sum_{i \in \mathcal{V}} \theta_i(x_i) + \sum_{(i,j) \in \mathcal{E}} \theta_{ij}(x_i, x_j). \tag{1}$$

The assignment variable x_i can assume values in $\{1, \dots, K\}$. The potentials $\theta_i(x_i)$ and $\theta_{ij}(x_i, x_j)$ encode unary and pairwise dependencies between the variables. For general graphs and energies the energy minimization in (1) is NP-hard. This problem arises in the context of finding the maximum-a-posteriori (MAP) prediction in Markov Random Fields. In this work we consider a fusion of a linear programming relaxation and a non-convex quadratic programming relaxation. As the quadratic part is introduced through a penalty function, the non-convexity of the objective can progressively be increased. Using the Kullback-Leibler divergence as the penalty function and a difference of convex functions method (DC), results in the standard LP objective with an additional pairwise entropy term for each edge. We solve the minimization problem using an efficient message-passing algorithm, which is repeatedly applied to modified unary potentials. We believe that this scheme combines the merits of the standard LP relaxation in terms of speed and scalability as well as the benefits of a compact description of the true constraint set. We demonstrate empirically that this leads to lower energy solutions than the ones obtained by LP solvers. We also compare our approach to a state of the art algorithm that further tightens the inexact LP relaxation.

2 Problem Formulation and Relaxation Approaches

The problem in (1) can be expressed as an integer quadratic program using K-ary coding:

$$\begin{aligned} \min_{\boldsymbol{\mu}} \quad & \sum_{i \in \mathcal{V}} \boldsymbol{\theta}_i^{\mathsf{T}} \boldsymbol{\mu}_i + \sum_{(i,j) \in \mathcal{E}} \boldsymbol{\mu}_i^{\mathsf{T}} \boldsymbol{\Theta}_{ij} \boldsymbol{\mu}_j \\ \text{s.t.} \quad & \mu_{i;k} \in \{0, 1\} \; \forall i, k \quad \text{and} \quad \sum_k \mu_{i;k} = 1 \; \forall i. \end{aligned} \tag{2}$$

We use the notation $\boldsymbol{\Theta}_{ij}$ to stress that the pairwise potentials here are represented in a matrix. Later we use $\boldsymbol{\theta}_{ij}$ to denote the vectorized version of this matrix, i.e. $\boldsymbol{\theta}_{ij} = \mathrm{vec}(\boldsymbol{\Theta}_{ij})$. An assignment of the variable x_i to state k corresponds to setting the k-th element of $\boldsymbol{\mu}_i$ (denoted by $\mu_{i;k}$) to one. Variational approaches for MAP inference can be divided according to whether the solution obtained is a lower bound or an upper bound on the energy of the true global minimum. The linear programming (LP) relaxation discussed next is an example of a lower bound. On the other hand the quadratic programming (QP) approach introduced in section 2.2, is an upper bound.

2.1 Linear Programming

The LP approach [1, 2] is based on a convex relaxation of (2). An additional variable $\boldsymbol{\mu}_{ij}$ is included for each edge. Proper marginalization is enforced through summation constraints. The resulting convex optimization problem is given by

$$\min_{\boldsymbol{\mu} \in \mathcal{L}_\mathcal{G}} \sum_{i \in \mathcal{V}} \boldsymbol{\theta}_i^\mathsf{T} \boldsymbol{\mu}_i + \sum_{(i,j) \in \mathcal{E}} \boldsymbol{\theta}_{ij}^\mathsf{T} \boldsymbol{\mu}_{ij}, \tag{3}$$

with $\mathcal{L}_\mathcal{G}$, the local marginal polytope:

$$\mathcal{L}_\mathcal{G} = \left\{ \boldsymbol{\mu} \;\middle|\; \begin{array}{ll} \sum_k \mu_{i;k} = 1 & \forall i \in \mathcal{V} \\ \sum_l \mu_{ij;kl} = \mu_{i;k} & \forall k, (i,j) \in \mathcal{E} \\ \sum_k \mu_{ij;kl} = \mu_{j;l} & \forall l, (i,j) \in \mathcal{E} \\ \mu_{ij;kl} \geq 0 & \forall k, l, (i,j) \in \mathcal{E} \end{array} \right\}. \tag{4}$$

In the general case $\mathcal{L}_\mathcal{G}$ is an inexact description of the marginal polytope, which requires an exponentially large number of constraints [2]. Including additional summation constraints over larger subsets of variables into $\mathcal{L}_\mathcal{G}$ tightens the LP relaxation further [3]. However, these methods generally suffer from an increased complexity as ultimately an exponentially large set of possible constraints needs to be searched over. An important property of LP-based approaches is their ability to give certificates of optimality. If the obtained solution is integer, the global optimum has been found.

2.2 Quadratic Programming

An alternative relaxation of the integer quadratic program in (2) is obtained by simply dropping the integer constraint. The resulting QP reads as follows:

$$\min_{\boldsymbol{\mu} \in \Delta_K^{|\mathcal{V}|}} \sum_{i \in \mathcal{V}} \boldsymbol{\theta}_i^\mathsf{T} \boldsymbol{\mu}_i + \sum_{(i,j) \in \mathcal{E}} \boldsymbol{\mu}_i^\mathsf{T} \boldsymbol{\Theta}_{ij} \boldsymbol{\mu}_j. \tag{5}$$

Here Δ_K denotes the simplex over K variables and $\Delta_K^{|\mathcal{V}|}$ the product space of $|\mathcal{V}|$ simplexes. Notice that the dimensionality of $\boldsymbol{\mu}$ is $K \cdot |\mathcal{V}|$, whereas in (3) the dimensionality is $K \cdot |\mathcal{V}| + K^2 \cdot |\mathcal{E}|$. A direct minimization of (5) is complex, as the product term for the edges makes the problem non-convex. There exist different approaches in the literature to deal with this non-convexity. [4] suggests convexifying the problem and [5] introduces a difference of convex (DC) functions approach. Finally, [6] recently introduced a message-passing algorithm for solving (5). The message-passing algorithm in our work is similar in spirit, as it ultimately solves the QP relaxation. However, our formulation also shares properties of the LP relaxation as it still includes the auxiliary pairwise variables. This helps overcoming poor local minima. In all of the experiments we conducted, the solution obtained by our formulation was at least comparable to the LP relaxation, most often substantially improving over it. This is generally not the case for QP relaxations, due to local minima.

3 Combining the LP and QP Relaxation

We consider a combination of the LP and QP relaxations. In our approach we keep the auxiliary variables $\boldsymbol{\mu}_{ij}$ for the pairwise terms, but force these variables to agree with the product of the unary marginals $\boldsymbol{\mu}_i$ and $\boldsymbol{\mu}_j$. The constraint is included as a penalty function in the objective, which allows us to enforce the constraint in a soft manner through a parameter β. As the penalty function we choose the Kullback-Leibler (KL) divergence. The combined objective reads as follows

$$\min_{\boldsymbol{\mu} \in \mathcal{L}_\mathcal{G}} \sum_{i \in \mathcal{V}} \boldsymbol{\theta}_i^\mathsf{T} \boldsymbol{\mu}_i + \sum_{(i,j) \in \mathcal{E}} \boldsymbol{\theta}_{ij}^\mathsf{T} \boldsymbol{\mu}_{ij} + \beta \sum_{(i,j) \in \mathcal{E}} D_{KL}(\boldsymbol{\mu}_{ij}, \mathrm{vec}(\boldsymbol{\mu}_i \boldsymbol{\mu}_j^\mathsf{T})). \tag{6}$$

Here $\mathrm{vec}(\boldsymbol{\mu}_i\boldsymbol{\mu}_j^\mathsf{T})$ denotes the vectorized version of the outer product of $\boldsymbol{\mu}_i$ and $\boldsymbol{\mu}_j$. Other distance functions, such as the squared Euclidean distance would be possible as well. We focus on the Kullback-Leibler divergence as it is the natural choice for distributions. The KL divergence for two discrete distributions $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ is given by

$$D_{KL}(\boldsymbol{\mu}, \boldsymbol{\nu}) = \sum_k \mu_k \log\left(\frac{\mu_k}{\nu_k}\right). \tag{7}$$

For $\beta = 0$ (6) reverts to the standard LP relaxation. For $\beta \to \infty$ the QP relaxation is reconstructed (and the auxiliary pairwise variables $\boldsymbol{\mu}_{ij}$ become redundant). Our MAP algorithm is based on successively increasing β and thereby slowly enforcing the constraint.

3.1 Difference of Convex Functions Decomposition

One way to solve a constrained optimization problem where the objective is non-convex, is through the *convex-concave procedure* (CCCP) [7], provided the objective has a decomposition into a convex and a concave part. We derive such a decomposition for the problem in (6). CCCP applied to the inference problem assumes a decomposition of the following form

$$\min_{\boldsymbol{\mu}\in\mathcal{L}_\mathcal{G}} u_\beta(\boldsymbol{\mu}) - v_\beta(\boldsymbol{\mu}), \tag{8}$$

where both, $u_\beta(\boldsymbol{\mu})$ and $v_\beta(\boldsymbol{\mu})$ are convex. For the combined relaxation problem in (6) a decomposition into a convex and concave function is given by

$$u_\beta(\boldsymbol{\mu}) = \sum_{i\in\mathcal{V}} \boldsymbol{\theta}_i^\mathsf{T}\boldsymbol{\mu}_i + \sum_{(i,j)\in\mathcal{E}} \boldsymbol{\theta}_{ij}^\mathsf{T}\boldsymbol{\mu}_{ij} - \beta \sum_{(i,j)\in\mathcal{E}} H(\boldsymbol{\mu}_{ij}) \tag{9}$$

$$v_\beta(\boldsymbol{\mu}) = -\beta \sum_{i\in\mathcal{V}} d_i H(\boldsymbol{\mu}_i). \tag{10}$$

Here d_i denotes the degree of node i and $H(\boldsymbol{\mu})$ denotes the entropy $H(\boldsymbol{\mu}) = -\sum_k \mu_k \log(\mu_k)$. In the derivation of the decomposition we made use of the fact that $-\sum_{k,l} \mu_{ij;kl} \log \mu_{i;k} = -\sum_{k,l} \mu_{i;k} \log \mu_{i;k}$ due to the marginalization constraints of the pairwise marginals. $u_\beta(\boldsymbol{\mu})$ consists of the original LP formulation with an additional term that promotes a high entropy on the pairwise marginal variables. $v_\beta(\boldsymbol{\mu})$ on the other hand induces a low entropy on the unary marginal variables $\boldsymbol{\mu}_i$. The CCCP algorithm proceeds by solving iteratively the convexified objective by linearizing the $v_\beta(\boldsymbol{\mu})$ term:

$$\boldsymbol{\mu}^{t+1} = \operatorname*{argmin}_{\boldsymbol{\mu}\in\mathcal{L}_\mathcal{G}} u_\beta(\boldsymbol{\mu}) - \boldsymbol{\mu}^\mathsf{T}\nabla v_\beta(\boldsymbol{\mu}^t). \tag{11}$$

The gradient of $v_\beta(\boldsymbol{\mu})$ is given by

$$\frac{\partial v_\beta(\boldsymbol{\mu})}{\partial \mu_{i;k}} = \beta d_i(1 + \log(\mu_{i;k})) \qquad \text{and} \qquad \frac{\partial v_\beta(\boldsymbol{\mu})}{\partial \mu_{ij;kl}} = 0. \tag{12}$$

Hence, the second term in (11) turns out to be an entropy approximation where the term $\log(\boldsymbol{\mu}_i)$ is replaced by $\log(\boldsymbol{\mu}_i^t)$, i.e. the marginal from the previous iteration is used instead (which is a constant).

3.2 The LPQP Algorithm

The full algorithm is shown in Algorithm 1. It consists of two loops, the inner loop solves the DC problem for a fixed β, the outer loop gradually increases the penalization parameter β. The main computational bottleneck is in line 5, where a particular convex optimization problem needs to be solved. As we will show in the next section, this turns out to be efficiently solved by a variant of the norm-product belief propagation algorithm. As the algorithm progresses and the β is increased, we would expect little changes in the solution and hence warm-starting the optimization problem on line 5 with the previous solution will lead to a massive speed increase. In all of the experiments we used $\beta_0 = 0.1$ and a multiplicative increase of β by a factor of 1.5. For a uniform initialization of the marginals, as used here, the first iteration does not depend on the initial solution and simply corresponds to solving the LP with an additional pairwise entropy term. We observed that the algorithm was less likely to return fractional solutions than the standard LP relaxation, but this could happen nevertheless. For the final solution we assigned each variable to the state with the largest marginal value.

Algorithm 1 The LPQP algorithm for MAP inference.

Require: $\mathcal{G} = (\mathcal{V}, \mathcal{E}), \boldsymbol{\theta}$.
initialize $\boldsymbol{\mu} \in \mathcal{L}_\mathcal{G}$ uniform, $\beta = \beta_0$.
repeat
  $t = 0, \boldsymbol{\mu}^0 = \boldsymbol{\mu}$.
  repeat
    $\boldsymbol{\mu}^{t+1} = \operatorname{argmin}_{\boldsymbol{\tau} \in \mathcal{L}_\mathcal{G}} u_\beta(\boldsymbol{\tau}) - \boldsymbol{\tau}^\mathsf{T} \nabla v_\beta(\boldsymbol{\mu}^t)$.
    $t = t + 1$.
  until $\|\boldsymbol{\mu}^t - \boldsymbol{\mu}^{t-1}\|_2 \leq \epsilon_{\text{dc}}$.
  $\boldsymbol{\mu} = \boldsymbol{\mu}^t$.
  increase β.
until $\|\boldsymbol{\mu} - \boldsymbol{\mu}^0\|_2 \leq \epsilon_\beta$.
return $\boldsymbol{\mu}$.

3.3 A Message Passing Algorithm for Solving the Convex Sub-problem

LPQP repeatedly solves the convex sub-problem for different penalty parameters β and unary potentials $\tilde{\boldsymbol{\theta}}_i$ using message-passing. The sub-problem is given by

$$\min_{\boldsymbol{\mu} \in \mathcal{L}_\mathcal{G}} \sum_{i \in \mathcal{V}} \tilde{\boldsymbol{\theta}}_i^\mathsf{T} \boldsymbol{\mu}_i + \sum_{(i,j) \in \mathcal{E}} \boldsymbol{\theta}_{ij}^\mathsf{T} \boldsymbol{\mu}_{ij} - \beta \sum_{(i,j) \in \mathcal{E}} H(\boldsymbol{\mu}_{ij}). \tag{13}$$

Here the modified unary potential

$$\tilde{\boldsymbol{\theta}}_i = \boldsymbol{\theta}_i - \beta d_i \log(\boldsymbol{\mu}_i^t), \tag{14}$$

corresponds to the original potential combined with the gradient term from the linearized part from the DC decomposition as in (11) [1]. It is interesting to observe that the potential aggressively discourages configurations that already had small probability in the previous iteration t.

The norm-product belief-propagation (BP) algorithm [8] solves (13). It is a generalization of (amongst others) belief-propagation [9] and tree-reweighted BP [10]. The algorithm is a primal-dual ascent algorithm and is guaranteed to converge to the global optimum for any choice of $\beta > 0$. In practice, we however noticed that for very small values of β numerical problems could occur and convergence was generally fairly slow. The norm-product algorithm applied to (13) computes messages passed from node j to node i as follows

$$m_{j \to i}(x_i) \propto \left(\sum_{x_j} \theta_{ij}^{1/\beta}(x_i, x_j) \frac{\theta_j^{1/(d_j \beta)}(x_j) \prod_{k \in \mathcal{N}(j)} m_{k \to j}^{1/(d_j \beta)}(x_j)}{m_{i \to j}^{1/\beta}(x_j)} \right)^\beta. \tag{15}$$

The marginals $\boldsymbol{\mu}_i$ are obtained by multiplying the incoming messages at variable i:

$$\mu_i(x_i) \propto \left(-\theta_i(x_i) \prod_{j \in \mathcal{N}(i)} m_{j \to i}(x_i) \right)^{1/(d_i \beta)}. \tag{16}$$

In our implementation of LPQP, the message-passing algorithm warm-starts using the messages from the previous DC iteration. Therefore, typically only few passes through the graph are needed to reach a convergence of the messages in later stages of the LPQP algorithm.

4 Experiments

We implemented Algorithm 1 in C++ and interfaced the code with Matlab using MEX. In our experiments we compare LPQP to LP relaxation solutions obtained using TRWS [11], a highly efficient message-passing algorithm for solving the LP relaxation. Furthermore, we compare also to solutions computed by MPLP [3] or if known, the global optimum. MPLP is a message-passing algorithm

[1] Notice that the βd_i part in ∇v_β is constant and can therefore be dropped.

that initially solves the LP relaxation but in later iterations also includes additional summation constraints over sets of three or four variables. MPLP was shown to identify the global optimum for some problems, but convergence is generally rather slow.

The output of the algorithms is an assignment of the variables, for which the energy in (1) can be computed and compared. However, the value of the energy obtained on a specific problem instance can not be used as a basis for comparison between instances. Moreover, the value of the energies lack a proper normalization for quantitative comparison between different results on the same instance. We therefore use the following energy comparison procedure. In each experiment run we set the worst and the best scores to zero and one respectively. The remaining score is then set to a fraction relative to its value between the best and the worst result. We use a linear scaling. This performance measure allows us to compare the algorithms as well as average over the results of different instances of the same task.

4.1 Synthetic Potts Model Data

We follow a similar experimental setup as in [12]. The graph is a 4-nearest neighbor grid of varying size. We used $M = 60, 90, 120$ where M is the side-length of the grid and M^2 is the overall number of variables. We used $K = 2$ and $K = 5$ for the number of states. The unary potentials were randomly set to $\theta_{i;k}(x_i) \sim \mathit{Uniform}(-\sigma, \sigma)$, we used σ values in $[0.05, 0.5]$. The pairwise potentials $\theta_{ij}(x_i, x_j)$ penalize agreements or disagreements of the labels by an amount chosen at random $\alpha_{i,j} \sim \mathit{Uniform}(-1, 1)$. Such that $\theta_{ij}(x_i, x_j) = 0$ if $x_i \neq x_j$ and $\alpha_{i,j}$ otherwise. The problem gets harder for small values of σ, the parameter can be understood as the signal-to-noise ratio.

We compared our algorithm with TRWS and MPLP using the performance measure described earlier. For each parameters choice we averaged the scores of 5 runs. The results are presented in Figure 1. As we expect TRWS returns the worst assignment on almost all configurations. In terms

M (size)	60		90		120	
K (# states)	2	5	2	5	2	5
			$\sigma = 0.05$			
MPLP	0.73	0.99	0.53	0.97	0	0.95
LPQP	1	0.99	1	1	1	1
TRWS	0	0	0	0	0.40	0
			$\sigma = 0.5$			
MPLP	1	1	1	1	1	1
LPQP	0.99	0.94	0.99	0.93	1	0.96
TRWS	0	0	0	0	0	0

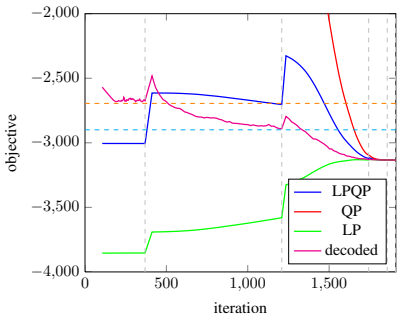

Figure 1: *Left*: Relative scores achieved by MPLP, LPQP and TRWS on the synthetic grid data. The scores are averaged over 5 runs, in each run the algorithms with the best and worst objective values get 1 and 0 respectively. The remaining algorithm gets a fractional score reflecting its relative objective value. *Right*: Development of the different objectives (for the same $\boldsymbol{\mu}$) during a run of the LPQP message-passing algorithm. The decoded objective refers to the current solution rounded to integer values. The vertical lines show iterations where β was increased. The orange and cyan horizontal dashed lines show the energy of the solution found by TRWS and MPLP, respectively.

of running time however, TRWS was always first to output a solution, followed by LPQP. MPLP was always slower and on the larger instances did not converge within a predefined maximal time. We therefore restricted the number of tightening iterations of MPLP to a maximum of 1000. A tightening iteration includes additional constraints into the local marginal polytope. Even after this change MPLP was still considerably slower than LPQP and TRWS. The energies obtained by LPQP and MPLP were very close on all configurations. We observe that the LPQP is doing better in comparison to the MPLP when the potentials are sampled with lower signal-to-noise ratio σ. This is also the case for a lower number of states, i.e. $K = 2$ instead of 5.

4.2 Protein Side-chain Prediction

As a real-world experiment we applied our MAP algorithm to the protein inference problem discussed in [13]. The data set was obtained from the supplementary material. It consists of two tasks: protein side-chain prediction and protein design. Here we consider the protein side-chain prediction problem. Earlier work [13], showed that only for 30 out of the 370 instances the LP relaxation is not tight. For 28 of the 30 instances, the true MAP has been computed in [13] using general integer programming techniques. We applied our algorithm to the 368 problem instances with known optimum.

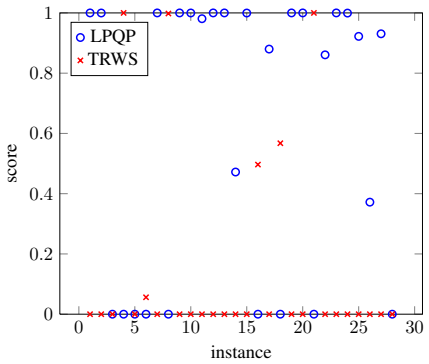

Figure 2: Protein prediction results. A score of one corresponds to finding the global minimum, a score of zero indicates the worse solution than (see text for more details). LPQP most often improves on TRWS, with a few exceptions. For around half of the problems the global optimum is found.

Figure 2 visualizes the results for the 28 instances where the LP relaxation is not tight. We used the same scoring as in the previous experiment only with the global optimum as the best result corresponding to a score of one. As before, the worst algorithm gets a score of zero. The LP relaxation (TRWS) performs worse than the LPQP algorithm. The LPQP finds the global minimum in roughly half of the instances. For the instances where the LP is tight (results are not shown), LPQP finds the global optimum in 75% of the cases. For the remaining 25% the energy sub-optimality is very small. MPLP was applied to the protein side-chain prediction dataset before [3] and found the global optimum in all instances. However, this comes at an increase in running time.

5 Conclusions and Future Work

We introduce a novel formulation for MAP inference in graphical models. The objective incorporates a quadratic constraint on the auxiliary pairwise LP variables. The quadratic constraint is enforced in a soft manner using as KL divergence penalty term in the objective. This helps overcoming poor local minima. A CCCP message-passing algorithm is derived to minimize the resulting objective. The LPQP algorithm shows substantial improvements over the standard LP relaxation. LPQP is also competitive when compared to LP relaxations that incorporate summation constraints over larger subsets of variables.

On the theoretical side the future work should include a more principled way to increment the penalty parameter β as well as an improved rounding scheme for obtaining final assignments. In addition we would like to investigate further the properties of the algorithm in terms of convergence guarantees. On the practical side we would like to apply LPQP to larger real-world problems, often encountered in computer vision. LPQP has a relatively small overhead when compared to standard message-passing algorithms such as TRWS, and is therefore expected to perform well on large-scale problems.

Acknowledgments

We would like to thank Christian Sigg, Andreas Krause and Joachim Buhmann for valuable input.

References

[1] M I Schlesinger. Syntactic analysis of two-dimensional visual signals in noisy conditions. *Kibernetika*, 4:113–130, 1976.

[2] Martin J Wainwright and Michael I Jordan. *Graphical Models, Exponential Families, and Variational Inference*, volume 1. 2008.

[3] David Sontag, Talya Meltzer, Amir Globerson, Yair Weiss, and Tommi Jaakkola. Tightening LP relaxations for MAP using message-passing. In *24th Conference in Uncertainty in Artificial Intelligence*, pages 503–510, 2008.

[4] Pradeep Ravikumar and John Lafferty. Quadratic Programming Relaxations for Metric Labeling and Markov Random Field MAP Estimation. In *ICML*, 2006.

[5] Joerg Kappes and Christoph Schnoerr. MAP-Inference for Highly-Connected Graphs with DC-Programming. In *DAGM*, 2008.

[6] Akshat Kumar and Shlomo Zilberstein. Message-Passing Algorithms for Quadratic Programming Formulations of MAP Estimation. In *UAI*, 2011.

[7] A L Yuille and Anand Rangarajan. The concave-convex procedure. *Neural Computation*, 15(4):915–36, 2003.

[8] Tamir Hazan and Amnon Shashua. Norm-product belief propagation: Primal-dual message-passing for approximate inference. *IEEE Transactions on Information Theory*, 56(12):6294–6316, 2010.

[9] Jonathan S. Yedidia, William T. Freeman, and Yair Weiss. Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Transactions on Information Theory*, 51(7):2282–2312, 2005.

[10] Martin J. Wainwright, Tommi Jaakkola, and Alan S. Willsky. MAP estimation via agreement on trees: message-passing and linear programming. *IEEE Transactions on Information Theory*, 51(11):3697–3717, 2005.

[11] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *PAMI*, 28(10):1568–83, 2006.

[12] Pradeep Ravikumar, Alekh Agarwal, and Martin J. Wainwright. Message-passing for graph-structured linear programs: Proximal methods and rounding schemes. *J. Mach. Learn. Res.*, 11:1043–1080, 2010.

[13] Chen Yanover, Talya Meltzer, and Yair Weiss. Linear programming relaxations and belief propagation – an empirical study. *J. Mach. Learn. Res.*, 7:1887–1907, 2006.