

Globally Convergent Dual MAP LP Relaxation Solvers using Fenchel-Young Margins

Authored by:

Tamir Hazan
Raquel Urtasun
Marc Pollefeys
Alex Schwing

Abstract

While finding the exact solution for the MAP inference problem is intractable for many real-world tasks, MAP LP relaxations have been shown to be very effective in practice. However, the most efficient methods that perform block coordinate descent can get stuck in sub-optimal points as they are not globally convergent. In this work we propose to augment these algorithms with an ϵ -descent approach and present a method to efficiently optimize for a descent direction in the subdifferential using a margin-based extension of the Fenchel-Young duality theorem. Furthermore, the presented approach provides a methodology to construct a primal optimal solution from its dual optimal counterpart. We demonstrate the efficiency of the presented approach on spin glass models and protein interactions problems and show that our approach outperforms state-of-the-art solvers.

1 Paper Body

Graphical models are a common method to describe the dependencies of a joint probability distribution over a set of discrete random variables. Finding the most likely configuration of a distribution defined by such a model, i.e., the maximum a-posteriori (MAP) assignment, is one of the most important inference tasks. Unfortunately, it is a computationally hard problem for many interesting applications. However, it has been shown that linear programming (LP) relaxations recover the MAP assignment in many cases of interest (e.g., [13, 23]). Due to the large amount of variables and constraints, solving inference problems in practice still remains a challenge for standard LP solvers. Development of specifically tailored algorithms has since become a growing area of research. Many of these designed solvers consider the dual program, thus they are based on local updates that follow the graphical model structure, which ensures suitability for

very large problems. Unfortunately, the dual program is non-smooth, hence introducing difficulties to existing solvers. For example, block coordinate descent algorithms, typically referred to as convex max-product, monotonically decrease the dual objective and converge very fast, but are not guaranteed to reach the global optimum of the dual program [3, 6, 11, 14, 17, 20, 22, 24, 25]. Different approaches to overcome the sub-optimality of the convex max-product introduced different perturbed programs for which convergence to the dual optimum is guaranteed, e.g., smoothing, proximal methods and augmented Lagrangian methods [6, 7, 8, 16, 18, 19, 27]. However, since these algorithms consider a perturbed program they are typically slower than the convex max-product variants [8, 18]. In this work we propose to augment the convex max-product algorithm with a steepest -descent approach to monotonically decrease the dual objective until reaching the global optimum of the dual program. To perform the -descent we explore the -subgradients of the dual program, and provide a method to search for a descent direction in the -subdifferential using a margin-based formulation of the Fenchel-Young duality theorem. This characterization also provides a new algorithm to

construct a primal optimal solution for the LP relaxation from a dual optimal solution. We demonstrate the effectiveness of our approach on spin glass models and protein-protein interactions taken from the probabilistic inference challenge (PIC 2011)¹. We illustrate that the method exhibits nice convergence properties while possessing optimality certificates. We begin by introducing the notation, MAP LP relaxations and their dual programs. We subsequently describe the subgradients of the dual and provide an efficient procedure to recover a primal optimal solution. We explore the -subgradients of the dual objective, and introduce an efficient globally convergent dual solver based on the -margin of the Fenchel-Young duality theorem. Finally, we extend our approach to graphical models over general region graphs.

2

Background

Graphical models encode joint distributions over discrete product spaces $X = X_1 \times \dots \times X_n$. The joint probability is defined by combining energy functions over subsets of variables. Throughout this work we consider two types of functions: single variable functions, $\phi_i(x_i)$, which correspond to the n vertices in the graph, $i \in \{1, \dots, n\}$, and functions over subsets of variables $\phi_E(x_E)$, for $E \subseteq \{1, \dots, n\}$, that correspond to the graph hyperedges. The joint distribution is then given by $p(x) \propto \exp(\sum_{i \in V} \phi_i(x_i) + \sum_{E \in \mathcal{E}} \phi_E(x_E))$. In this paper we focus on estimating the MAP, i.e., finding the assignment that maximizes the probability, or equivalently minimizes the energy which is the negative log probability. Estimating the MAP can be written as a program of the form [10]: $\max_{x_1, \dots, x_n} \sum_{i \in V} \phi_i(x_i) + \sum_{E \in \mathcal{E}} \phi_E(x_E)$.

$\sum_{i \in V}$
 $\sum_{E \in \mathcal{E}}$

Due to its combinatorial nature, this problem is NP-hard for general graphical models. It is tractable only in some special cases such as tree structured graphs, where specialized dynamic programming algorithms (e.g., max-product

belief propagation) are guaranteed to recover the optimum. The MAP program in Eq. (1) has a linear form, thus it is naturally represented as an integer linear program. Its tractable relaxation is obtained by replacing the integral constraints with non-negativity constraints as follows:

$$\begin{aligned} & \max_{\mathbf{x}, \mathbf{b}} \sum_i b_i(x_i) \\ & \text{s.t. } b_i(x_i), b_e(x_e) \geq 0, \\ & \sum_{x_e} b_e(x_e) = 1, \\ & \sum_{x_i} b_i(x_i) = 1, \\ & b_e(x_e) = b_i(x_i). \end{aligned}$$

Whenever the maximizing argument to above linear program happens to be integral, i.e., the optimal beliefs satisfy $b_i(x_i), b_e(x_e) \in \{0, 1\}$, the program value equals the MAP value. Moreover, the maximum arguments of the optimal beliefs point toward the MAP assignment [26]. We denote by $N(i)$ the edges that contain vertex i and by $N(e)$ the vertices in the edge e . Following [22, 27] we consider the re-parametrized dual

$$\begin{aligned} & \max_{\mathbf{q}} \sum_i q_i \\ & \text{s.t. } q_i \geq 0 \end{aligned}$$

The dual program value upper bounds the primal program described in Eq. (2). Therefore to compute the primal optimal value one can minimize the dual upper bound. Using block coordinate descent on the dual objective amounts to optimizing blocks of dual variables while holding the remaining ones fixed. This results in the convex max-product message-passing update rules [6, 17]: Repeat until convergence, for every $i = 1, \dots, n$:

$$\begin{aligned} & q_i \leftarrow \max_{x_i} \left(\sum_{e \in N(i)} q_e(x_e) + \sum_{j \in N(e)} q_j(x_j) \right) \\ & q_i \leftarrow \max_{x_i} \left(\sum_{e \in N(i)} q_e(x_e) + \sum_{j \in N(e)} q_j(x_j) \right) \end{aligned}$$

$$\begin{aligned} & \text{http://www.cs.huji.ac.il/project/PASCAL/index.php} \\ & 2 \end{aligned}$$

The convex max-product algorithm is guaranteed to converge since it minimizes the dual function, which is lower bounded by the primal program. Interestingly, the convex max-product shares the same complexity as the max-product belief propagation, which is attained by replacing the coefficient $1/(1$

+ —N (i)— by 1. It has, however, two fundamental problems. First, it can get stuck in non-optimal stationary points. This happens since the dual objective is non-smooth, thus the algorithm can reach a corner, for which the dual objective stays fixed when changing only a few variables. For example, consider the case of a minimization problem where we try to descend from a pyramid while taking only horizontal and vertical paths. We eventually stay at the same height. The second drawback of convex max-product is that it does not always produce a primal optimal solution, $b_i(x_i)$, $b_j(x_j)$, even when it reaches a dual optimal solution. In the next section, we consider the dual subgradients, and provide an efficient algorithm for detecting corners, as well as for decoding a primal optimal solution from a dual optimal solution. This is an intermediate step which facilitates the margin analysis of the Fenchel-Young duality theorem in Sec. 4. It provides an efficient way to get out of corners, and to reach the optimal dual value.

3

The Subgradients of the Dual Objective and Steepest Descent

Subgradients are generalizations of gradients for non-smooth convex functions. Consider the function $q(\cdot)$ in Eq. (3). A vector d is called a subgradient of $q(\cdot)$ if it supports the epigraph of $q(\cdot)$ at \cdot , i.e., $q(\cdot) \geq d \cdot \cdot$. (4) The supporting hyperplane at $(\cdot, q(\cdot))$ with slope d takes the form $d \cdot \cdot \leq q(\cdot)$, when defining the conjugate dual as $q^*(d) = \max_{\cdot} \{d \cdot \cdot - q(\cdot)\}$. From the definition of $q^*(d)$ one can derive the Fenchel-Young duality theorem: $q(\cdot) + q^*(d) \geq d \cdot \cdot$, where equality holds if and only if d is a supporting hyperplane at $(\cdot, q(\cdot))$. The set of all subgradients is called the subdifferential, denoted by $\partial q(\cdot)$, which can be characterized using the Fenchel-Young theorem as $\partial q(\cdot) = \{d : q(\cdot) + q^*(d) = d \cdot \cdot\}$. The subdifferential provides a way to reason about the optimal solutions of $q(\cdot)$. Using Eq. (4) we can verify that \cdot is dual optimal if and only if $0 \in \partial q(\cdot)$. In the following claim we characterize the subdifferential of the dual function $q^*(\cdot)$ using the Fenchel-Young duality theorem: Claim 1. Consider the dual function $q^*(\cdot)$ given in Eq. (3). Let $X_i = \arg\max_{x_i} \{q_i(x_i) - \sum_{j \neq i} b_j(x_j) + i \sum_{j \neq i} q_j(x_j)\}$. Then $d \in \partial q^*(\cdot)$, if $\sum_{i \in N} d_i q_i(x_i) = \sum_{i \in N} b_i(x_i)$ for probability distributions $b_i(x_i)$, $b_j(x_j)$ whose nonzero entries belong to X_i , X_j respectively. Proof: Using the Fenchel-Young characterization of Eq. (4) for the max-function we obtain the set of maximizing elements X_i , X_j . Summing over all regions $r \in \{i, j\}$ while noticing the change of sign, we obtain the marginalization disagreements $d_i q_i(x_i)$. The convex max-product algorithm performs block coordinate descent updates. Thus it iterates over vertices i and computes optimal solutions $q_i(x_i)$ for every x_i , $\cdot \in N(i)$ analytically, while holding the rest of the variables fixed. The claim above implies that the convex max-product iterates over i and generates beliefs $b_i(x_i)$, $b_j(x_j)$ for every x_i , $\cdot \in N(i)$ that agree on their marginal probabilities. This interpretation provides an insight into the non-optimal stationary points of the convex max-product, i.e., points for which it is not able to generate consistent beliefs $b_j(x_j)$ when it iterates over $i = 1, \dots, n$. The representation of the subdifferential as the amount of disagree-

ment between the marginalization constraints provides a simple procedure to verify dual optimality, as well as to construct primal optimal solutions. This is summarized in the corollary below. Corollary 1. Given a point β , and sets X_i , X as defined in Claim 1, let x_i , x be elements in X_i , X respectively. Consider the quadratic program

$$\begin{aligned} & \min_{x_i, x} \beta^T (x - x_i) \\ & \text{s.t. } \beta^T (x_i - x) \leq 0, \\ & \quad \beta^T (x - x_i) = 1, \\ & \quad \beta^T (x_i - x) = 1. \end{aligned}$$

β is a dual optimal solution if and only if the value of the above program equals zero. Moreover, if β is a dual optimal solution, then the optimal beliefs $\beta_i(x_i)$, $\beta(x)$ are also the optimal solution

of program in Eq. (2). However, if β is not dual optimal, then the vector $d_i(x_i) = P$ the primal $\beta^T (x - x_i) \leq 0$ points towards the steepest descent direction of the dual function, i.e., $x - x_i$

$$\begin{aligned} d_i &= \arg\min_{x} \|x - x_i\| \\ \lim_{\epsilon \rightarrow 0} \beta^T (x - x_i) &= 0 \end{aligned}$$

Proof: The steepest descent direction d of q is given by minimizing the directional derivative $\nabla q(d)$, $\min_{\|d\|=1} \nabla q(d) = \min_{\|d\|=1} \max_{\|y\|=1} d^T y = \max_{\|y\|=1} \min_{\|d\|=1} d^T y =$

$$\begin{aligned} & \min_{\|d\|=1} \max_{\|y\|=1} d^T y \\ & \max_{\|y\|=1} \min_{\|d\|=1} d^T y \end{aligned}$$

which yields the above program (cf. [2], Chapter 4). If the zero vector is part of the subdifferential, we are dual optimal. Primal optimality follows from Claim 1. One can monotonically decrease the dual objective by minimizing it along the steepest descent direction. Unfortunately, following the steepest descent direction does not guarantee convergence to the global minimum of the dual function [28]. Performing steepest descent might keep minimizing the dual objective with smaller and smaller increments, thus converging to a suboptimal solution. The main drawback of steepest descent as well as block coordinate descent when applied to the dual objective in Eq. (3) is that both procedures only consider the support of X_i , X defined in Claim 1. In the following we show that by considering the ϵ -margin of these supports we can guarantee that at every iteration we decrease the dual value by at least ϵ . This procedure results in an efficient algorithm that reaches both dual and primal optimal solutions.

4

The ϵ -Subgradients of the Dual Objective and Steepest ϵ -Descent

$x_i b_i(x_i) \leq b_i(x_i)$. $q(x)$ is described using beliefs $b_i(x_i)$, $b_i(x_i)$ that satisfy linear constraints, therefore finding a direction of -descent can be done efficiently. Claim 2 ensures that minimizing the dual objective along a direction of descent decreases its value by at least ϵ . Moreover, we are guaranteed to be $(-V - \epsilon - E)$ -close to a dual optimal solution if no direction of descent is found in $q(x)$. Therefore, we are able to get out of corners and efficiently reach an approximated dual optimal solution. The interpretation of the Fenchel-Young margin as the amount of disagreement between the marginalization constraints also provides a simple way to reconstruct an approximately optimal primal solution. This is summarized in the following corollary. \square Corollary 2. Given a point x , set $b_i(x_i) = b_i(x_i) \leq b_i(x_i)$ and $b_i(x_i) = b_i(x_i) + b_i(x_i)$. Consider the quadratic program $\min_{b_i(x_i)} b_i(x_i)$

$$\begin{aligned} & \min_{b_i(x_i)} b_i(x_i) \\ & \text{s.t. } b_i(x_i), b_i(x_i) \geq 0, \\ & \quad b_i(x_i) = 1, \\ & \quad b_i(x_i) = 1 \\ & \quad b_i(x_i) \leq \max\{b_i(x_i)\} \leq x_i \\ & \quad b_i(x_i) \leq \max\{b_i(x_i)\} \leq x_i \end{aligned}$$

$q(x)$ is $(-V - \epsilon - E)$ -close to the dual optimal value if and only if the value of the above program equals zero. Moreover, the optimal beliefs $b_i(x_i)$, $b_i(x_i)$ primal value is $(-V - \epsilon - E)$ -close to the optimal primal value in Eq. (2). \square However, if $q(x)$ is not $(-V - \epsilon - E)$ -close to the dual optimal value then the vector $d_i(x_i) = x_i - b_i(x_i)$ points towards the steepest -descent direction of the function, namely $q(x) + d_i(x_i) = \arg\min_{d_i(x_i)} \|d_i(x_i)\|$. Proof: The steepest -descent direction is given by the minimum norm element of the subdifferential, described in Claim 3. $(-V - \epsilon - E)$ -closeness to the dual optimum is given by ([2], Proposition 4.3.1) once we find the value of the quadratic program to be zero. Note that the superset \mathcal{Q} is composed of $-V - \epsilon - E$ subdifferentials. If the value of the above program equals zero, the beliefs fulfill marginalization constraints and they denote a probability distribution. Summing both -margin inequalities w.r.t. i , we obtain $\sum_i b_i(x_i) \leq \sum_i b_i(x_i) + b_i(x_i) \leq \max_i b_i(x_i) + \max_i b_i(x_i) \leq (-V - \epsilon - E)$. \square

$$\begin{aligned} & \min_{b_i(x_i)} b_i(x_i) \\ & \quad b_i(x_i) = 1 \\ & \quad b_i(x_i) = 1 \end{aligned}$$

?

x?

where the primal on the left hand side of the resulting inequality is larger than the dual subtracted by $(-V - + -E-)$. With the dual itself upper bounding the primal, the corollary follows. Thus, we can construct an algorithm that performs improvements over the dual function in each iteration. We can either perform block-coordinate dual descent (i.e., convex max-product updates) or steepest -descent steps. Since both methods monotonically improve the same dual function, our approach is guaranteed to reach the optimal dual solution and to recover the primal optimal solution. 5

(a)

(b)

Figure 1: (a) Difference between the minimal dual value attained by convex max-product $q(\text{CMP})$ and our approach $q(\text{ })$. Convex max-product gets stuck in about 20% of all cases. (b) Dual value achieved after a certain amount of time for cases where convex max-product gets stuck.

5

High-Order Region Graphs

Graphical models naturally describe probability distributions with different types of regions $r \in \{1, \dots, n\}$. However, the linear program relaxation described in Eq. (2) considers interactions between regions which correspond to variables i and regions that correspond to cliques τ . In the following we extend the -descent framework when considering linear programming relaxations without constraining the region interactions. Since we allow any regions to interact, we denote these interactions through a region graph [29]. A region graph is a directed graph whose nodes represent the regions and its direct edges correspond to the inclusion relation, i.e., a directed edge from node r to s is possible only if $s \subseteq r$. We adopt the terminology where $P(r)$ and $C(r)$ stand for all nodes that are parents and children of the node r , respectively. Thus we consider the linear programming relaxation of a general high-order graphical model as follows

$$\begin{aligned} & \max_{\mathbf{x}} \sum_{r \in \mathcal{R}} \mathbf{c}_r(\mathbf{x}_r) \\ & \text{s.t. } \mathbf{b}_r(\mathbf{x}_r) \leq 0, \\ & \mathbf{x}_r \in \mathcal{X}_r \\ & \mathbf{x}_s \in \mathcal{X}_s \text{ for } s \in P(r) \\ & \mathbf{b}_r(\mathbf{x}_r) = 1, \\ & \mathbf{x}_r \in \mathcal{X}_r \\ & \mathbf{b}_s(\mathbf{x}_s) = \mathbf{b}_r(\mathbf{x}_r) \end{aligned}$$

Following [5, 22, 27] we consider the re-parametrized dual program

$$\begin{aligned} q(\text{ }) = \max_{\mathbf{c}, \mathbf{p}} & \sum_{r \in \mathcal{R}} \mathbf{c}_r(\mathbf{x}_r) + \sum_{r \in \mathcal{R}} \mathbf{p}_r(\mathbf{x}_r) \\ & \text{s.t. } \mathbf{c}_r \in C(r) \\ & \mathbf{p}_r \in P(r) \end{aligned}$$

which is a sum of max-functions. Its approximated ϵ -subdifferential is described with respect to their Fenchel-Young margins. Using the same reasoning as in Sec. 4 we present a simple way to recover an ϵ -steepest descent direction, as well as to reconstruct an approximated optimal primal solution. \square **Corollary 3.** Given a point x , set $\tilde{r}(x) = r(x) + \epsilon \nabla r(x) - \nabla p(x)$. $\epsilon \in C(r)$

$\tilde{p}(r)$
Consider the quadratic program $\min_{x \in X} b(x) - \tilde{r}(x)$, $x \in X, \tilde{p}(r)$
(r)

s.t.
 $b(x) \geq 0$,
 $x \in X$
 $x \in X$
 $b(x) = 1$,
 $x \in X$
 $b(x) - \tilde{r}(x) \geq \max\{\tilde{r}(x)\}$
 $x \in X$
 $x \in X$

Let R be the total number of regions in the graph, then \tilde{r} is R -close to the dual optimal solution if and only if the value of the above program equals zero. Moreover, the optimal beliefs $\tilde{r}(x)$ are also R -close to the optimal solution of the primal program in Eq. (7). However, if $q(\tilde{r})$ is not R -close to the dual optimal solution then the vector $\tilde{p}(x) = x - \tilde{r}(x)$ points towards the steepest ϵ -descent direction of the dual function. Proof: It is a straightforward generalization of Corollary 2. When dealing with high-order region graphs, one can choose a region graph, e.g., the Hasse diagram, that has significantly less edges than a region graph that connects variables i to cliques C . Therefore, when considering many high-order regions, the formulation in the above corollary is more efficient than the one in Corollary 2. \square

time [s]
time [s]
(a)
(b)
(c)

Figure 2: Average time required for different solvers to achieve a specified accuracy on 30 spin glass models, (a) when solvers are applied to ‘hard’ problems only, i.e., those where CMP gets stuck far from the optimum. Average results over 30 models are shown in (b), (c) decrease of the dual value over time for ADLP and our ϵ -descent approach.

6

Experimental Evaluation

To benefit from the efficiency of convex max-product, our implementation starts by employing block-coordinate descent iterations before switching to the globally convergent ϵ -descent approach once the dual value decreases by less than $\epsilon = 0.01$. As we always optimize the same cost function, switching the gradient computation is possible. We employ a backtracking line search in our ϵ -descent

approach. In the following we demonstrate the effectiveness of our approach on synthetic 10x10 spin glass models as well as protein interactions from the probabilistic inference challenge (PIC 2011). We consider spin glass models that consist of local factors, each having 3 states with values randomly chosen according to $N(0, 1)$. We use three states as convex max-product is optimal for pairwise spin glass models with only two states per random variable. The pairwise factors of the regular grid are weighted potentials with +1 on the diagonal and off-diagonal entries being ± 1 . The weights are again independently drawn from $N(0, 1)$. In the first experiment we are interested in estimating how often convex max-product gets stuck in corners. We generate a set of 1000 spin glass models and estimate the distribution of the dual value difference comparing the ℓ_1 -descent approach with the convex max-product result after 10,000 iterations. We observe in Fig. 1(a) that about 20% of the spin glass models have a dual value difference larger than zero. Having observed that convex max-product does not achieve optimality for 20% of the models, we now turn our attention to evaluating the run-time of different algorithms. We compare our implementation of the ℓ_1 -steepest descent algorithm with the alternating direction method for dual MAPLP relaxations (ADLP) [18]. In addition, we illustrate the performance of convex max-product (CMP) [6] and compare against the dual-decomposition work of [12] provided in a generic (DDG) and a re-weighted (DDR) version in the STAIR library [4]. Note that ADLP is also implemented in this library. All algorithms are restricted to at most 20,000 iterations. We draw the readers attention to Fig. 1(b), where we evaluate a single spin glass model and illustrate the dual value obtained after a certain amount of time. As given by the derivations, CMP is a monotonically decreasing algorithm that can get stuck in corners. It is important to note that our ℓ_1 -descent approach is monotonically decreasing as well, which contrasts all the other investigated algorithms (ADLP, DDG, DDR). We evaluate the time it requires the different algorithms to achieve a given accuracy. We first focus on "hard" problems, where we defined "hard" as those spin glass models whose difference between convex max-product and the ℓ_1 -descent method is larger than 0.2. To obtain statistically meaningful results we average over 30 hard problems and report the time to achieve a given accuracy in Fig. 2(a). We used the minimum across all dual values found by all algorithms as the optimum. If an algorithm does not achieve ϵ -close accuracy within 20,000 iterations we set its time to the arbitrarily chosen value of 10^5 . We note that CMP is very fast for low accuracies (high ϵ) but gets stuck in corners, not achieving high accuracies (low ϵ). This is also the case for DDG and DDR. ADLP achieves significantly lower ϵ -closeness but the 20,000 iteration limit stops it from reaching 10^{-3} . The previous experiment focus on hard problems. In order to evaluate the average case, we randomly generate 30 spin glass models. The results are provided in Fig. 2(b). As expected the ℓ_1 -descent approach performs similarly well, ADLP achieves lower accuracies on more samples. The step apparent for CMP, DDG and DDR is not as sharp, but still very significant. Protein interactions: We rely on the data provided by the PIC 2011 and compare the ℓ_1 -descent approach to ADLP as it is the most competitive method in the previous experiments. The dual energy obtained

after a given amount of time is illustrated in Fig. 2(c). 7

7

Related Work

We explore methods to solve LP relaxations by monotonically decreasing the value of its dual objective and reconstructing a primal optimal solution. For this purpose we investigate approximated subgradients of the dual program using the Fenchel-Young margins, and provide a method to reduce the dual objective in every step by a constant value until convergence to the optimum. Efficient dual solvers were extensively studied in the context of LP relaxations for the MAP problem [14, 20, 25]. The dual program is non-smooth, thus subgradient descent algorithms are guaranteed to reach the dual optimum [12], as well as recover the primal optimum [12]. Despite their theoretical guarantees, subgradient methods are typically slow. Dual block coordinate descent methods, typically referred to as convex max-product algorithms, are monotonically decreasing, and were shown to be faster than subgradient methods [3, 6, 11, 17, 22, 24, 27]. Since the dual program is non-smooth, these algorithms can get stuck in non-optimal stationary points and cannot in general recover a primal optimal solution [26]. Our work specifically addresses these drawbacks. Recently, several methods were devised to overcome the sub-optimality of convex max-product algorithms. Unlike our approach, all these algorithms optimize a perturbed program. Some methods use the soft-max with low temperature to smooth the dual objective in order to avoid corners as well as to recover primal optimal solutions [6, 7, 8]. However, these methods are typically slower, as computation of the low-temperature soft-max is more expensive than max-product. [19] applied the proximal method, employing a primal strictly concave perturbation, which results in a smooth dual approximation that is temperature independent. This approach converges to the dual optimum and recovers the primal optimal solution. However, it uses a double loop scheme where every update involves executing a convex sum-product algorithm. Alternative methods applied augmented Lagrangian techniques to the primal [16] and the dual programs [18]. The augmented Lagrangian method guarantees to reach the global optimum and recover the dual and primal solutions. Unlike our approach, this method is not monotonically decreasing and works on a perturbed objective, thus cannot be efficiently integrated with convex max-product updates that perform block coordinate descent on the dual of the LP relaxation. Our approach is based on the γ -descent algorithm for convex functions [2]. We use the γ -margin of the Fenchel-Young duality theorem to adjust the γ -subdifferential to the dual objective of the LP relaxation, thus augmenting the convex max-product with the ability to get out of corners. We also construct an efficient method to recover a primal optimal solution. Our approach is related to the Bundle method [15, 9], which performs an γ -subgradient descent in cases where efficient search in the γ -subdifferential is impossible. The graphical model structure in our setting makes searching in the γ -subdifferential easy, thus our approach is significantly faster. Our algorithm satisfies γ -complementary slackness while performing γ -descent step, similarly to the auction algorithm. However, our algorithm is monotonically decreasing and can be used for general graphical

models, while the auction algorithm might increase its dual and its convergence properties hold only for network flow problems.

8

Conclusions and Discussion

Evaluating the MAP assignment and solving its LP relaxations are key problems in approximate inference. Some of the existing solvers, such as convex max-product, have limitations. Mainly, these solvers can get stuck in a non-optimal stationary point, thus they cannot recover the primal optimal solution. We explore the properties of subgradients of the dual objective and construct a simple algorithm that determines if the dual stationary point is optimal and recovers the primal optimal solution in this case (Corollary 1). Moreover, we investigate the family of ϵ -subgradients using Fenchel-Young margins and construct a monotonically decreasing algorithm that is guaranteed to achieve optimal dual and primal solutions (Corollary 2), including general region graphs (Corollary 3). We show that our algorithm compares favorably with previous methods on spin glass models and protein interactions. The approximated steepest descent direction is recovered by solving a quadratic program subject to linear constraints. We used the Gurobi solver2, which ignores the graphical structure of the linear constraints. We believe that constructing a message-passing solver for this sub-problem will significantly speed-up our approach. Further extensions, e.g., enforcing constraints over messages such as those arising from cloud computing are also applicable to our setting [1, 21]. 2

<http://www.gurobi.com>

8

2 References

- [1] A. Auslender and M. Teboulle. Interior gradient and epsilon-subgradient descent methods for constrained convex minimization. Mathematics of Operations Research, 2004.
- [2] D. P. Bertsekas, A. Nedić, and A. E. Ozdaglar. Convex Analysis and Optimization. Athena Scientific, 2003.
- [3] A. Globerson and T. S. Jaakkola. Fixing max-product: convergent message passing algorithms for MAP relaxations. In Proc. NIPS, 2007.
- [4] S. Gould, O. Russakovsky, I. Goodfellow, P. Baumstarck, A. Y. Ng, and D. Koller. The STAIR Vision Library (v2.4), 2011. <http://ai.stanford.edu/~sgould/svl>.
- [5] T. Hazan, J. Peng, and A. Shashua. Tightening fractional covering upper bounds on the partition function for high-order region graphs. In Proc. UAI, 2012.
- [6] T. Hazan and A. Shashua. Norm-product belief propagation: Primal-dual message-passing for approximate inference. Trans. on Information Theory, 2010.
- [7] J. K. Johnson. Convex relaxation methods for graphical models: Lagrangian and maximum entropy approaches. PhD thesis, Massachusetts Institute of Technology, 2008.
- [8] V. Jojić, S. Gould, and D. Koller. Accelerated dual decomposition for MAP inference. In Proc. ICML, 2010.
- [9] J. H. Kappes, B. Savchynskyy, and C. Schnörr. A Bundle Approach To Efficient MAP-Inference by Lagrangian Relaxation. In Proc. CVPR, 2012.
- [10] D. Koller and N. Friedman. Prob-

bilistic graphical models. MIT Press, 2009. [11] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. PAMI, 2006. [12] N. Komodakis, N. Paragios, and G. Tziritas. MRF Energy Minimization & Beyond via Dual Decomposition. PAMI, 2010. [13] T. Koo, A.M. Rush, M. Collins, T. Jaakkola, and D. Sontag. Dual decomposition for parsing with nonprojective head automata. In Proc. EMNLP, 2010. [14] A.M.C.A. Koster, S.P.M. van Hoesel, and A.W.J. Kolen. The partial constraint satisfaction problem: Facets and lifting theorems. Operations Research Letters, 1998. [15] C. Lemaréchal. An algorithm for minimizing convex functions. Information processing, 1974. [16] A.F.T. Martins, M.A.T. Figueiredo, P.M.Q. Aguiar, N.A. Smith, and E.P. Xing. An Augmented Lagrangian Approach to Constrained MAP Inference. In Proc. ICML, 2011. [17] T. Meltzer, A. Globerson, and Y. Weiss. Convergent Message Passing Algorithms ? A Unifying View. In Proc. UAI, 2009. [18] O. Meshi and A. Globerson. An Alternating Direction Method for Dual MAP LP Relaxation. In Proc. ECML PKDD, 2011. [19] P. Ravikumar, A. Agarwal, and M. J. Wainwright. Message-passing for graph-structured linear programs: Proximal methods and rounding schemes. JMLR, 2010. [20] M. Schlesinger. Syntactic analysis of two-dimensional visual signals in noisy conditions. Kibernetika, 76. [21] A. G. Schwing, T. Hazan, M. Pollefeys, and R. Urtasun. Distributed message passing for large scale graphical models. In Proc. CVPR, 2011. [22] D. Sontag and T. S. Jaakkola. Tree block coordinate descent for MAP in graphical models. In Proc. AISTATS, 2009. [23] D. Sontag, T. Meltzer, A. Globerson, T. Jaakkola, and Y. Weiss. Tightening LP relaxations for MAP using message passing. In Proc. UAI, 2008. [24] D. Tarlow, D. Batra, P. Kohli, and V. Kolmogorov. Dynamic tree block coordinate ascent. In Proc. ICML, 2011. [25] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. MAP estimation via agreement on trees: messagepassing and linear programming. Trans. on Information Theory, 2005. [26] Y. Weiss, C. Yanover, and T. Meltzer. MAP Estimation, Linear Programming and Belief Propagation with Convex Free Energies. In Proc. UAI, 2007. [27] T. Werner. Revisiting the linear programming relaxation approach to gibbs energy minimization and weighted constraint satisfaction. PAMI, 2010. [28] P. Wolfe. A method of conjugate subgradients for minimizing nondifferentiable functions. Nondifferentiable Optimization, 1975. [29] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Constructing free-energy approximations and generalized belief propagation algorithms. Trans. on Information Theory, 2005.