SDCA-Powered Inexact Dual Augmented Lagrangian Method for Fast CRF Learning

Author 1
Institution 1

Author 2
Institution 2

Author 3
Institution 3

Abstract

We propose an efficient *dual* augmented Lagrangian formulation for conditional random fields (CRF) learning. Our algorithm, which can be interpreted as an inexact gradient method, does not require to perform global inference iteratively, requires only a fixed number of stochastic clique-wise updates at each epoch to obtain a sufficiently good estimate of the gradient for the Lagrangian multipliers. We prove that the proposed algorithm enjoys global linear convergence in both the primal and the dual objectives. Our experiments show that the proposed algorithm outperforms state-of-the-art baselines in terms of the speed of convergence.

1 Introduction

Learning in graphical models has historically relied on the computation of the (sub)gradient of model parameter that requires to solve a MAP or probabilistic inference problem at each iteration. This alternating approach is slow given that the inference problem itself is computationally expensive. The difficulty of inference and learning in graphical models is related to the fact that the log-partition function and its gradient are in general intractable.

An optimization problem whose objective is a large finite sum of convex terms have been shown to be optimized very efficiently by stochastic algorithms that sample one term at a time (Schmidt et al., 2013; Shalev-Shwartz and Zhang, 2014; Defazio et al., 2014).

The dual problem of the maximum likelihood estimation of CRF (a.k.a. the maximum entropy principle) decomposes additively over all cliques if a decomposable entropy surrogate is adopted. Even though this dual formulation has a potential to take advantage of stochastic algorithms, and can be optimized without resorting to solve a global inference on the entire graph per iteration, all dual parameters (i.e. *mean parameters*) are constrained by the *marginal polytope*, which is in general an intractable polytope. Even its most commonly used relaxation, namely the *local consistency polytope*, is itself in practice difficult to optimize over. Recently, Meshi et al. (2015b,a) proposed to replace the marginalization constraints, which are part of the local consistency polytope, with quadratic penalty terms. The relaxed problem has then only separable constraints over the cliques that makes it possible to use efficient block coordinate optimization schemes.

Following these ideas, we consider a dual formulation for CRF learning in which the marginalization constraints are replaced by an augmented Lagrangian term, and at meanwhile the intractable Shannon entropy is replaced by a smooth and strongly concave surrogate, e.g. the Gini entropy, so that stochastic dual coordinate ascent (SDCA) can be used to optimize with respect to the mean parameters, with similar guarantees as in Shalev-Shwartz and Zhang (2014). We finally show that by periodically updating the Lagrangian multipliers as we are optimizing the relaxed dual, we can gradually enforce the marginalization constraints, while retaining global linear convergence. In terms of the primal problem associated with the Lagrange multipliers, our algorithm is an inexact gradient descent algorithm using stochastic approximation of the multiplier gradients.

Our paper is organized as follows. We review CRF learning in Section 3. A dual augmented Lagrangian formulation is presented in Section 4. The proposed algorithm is presented in Section 5, followed by its convergence analysis in Section 6. Finally, we present experiments on three applications in Section 7 (Most notations used in the paper can be found in Appendix F).

Preliminary work. Under review by AISTATS 2018. Do not distribute.

2 Related Work

Due to the independent interest of inference problem in discrete graphical models, in particular for computer vision applications, a significant amount of work has been devoted to develop efficient approximate inference algorithms (Komodakis et al., 2007; Sontag et al., 2008; Savchynskyy et al., 2011; Martins et al., 2015). However, the learning problem is not necessarily easier (can even fail to converge) with an approximate inference approach as the subroutine (Kulesza and Pereira, 2007).

There is a large body of research on efficient algorithms of structured learning. For the max-margin formulation, the fastest algorithms to date rely on block coordinate Frank-Wolfe updates (Lacoste-Julien et al., 2013; Meshi et al., 2015b; Tang et al., 2016). Using dual decomposition in the inner inference problem, Meshi et al. (2010); Hazan and Urtasun (2010); Komodakis (2011) proposed to solve the classical saddle-point formulation for structured learning problem with algorithms that alternate between message passing and the model parameter updates. Going further Meshi et al. (2015b); Yen et al. (2016) work on a purely dual formulation to enable clique-wise updates. For maximum likelihood learning, exponentiated gradient and its block variants can be applied (Collins et al., 2008). Other recent work have relied on incremental algorithms (Schmidt et al., 2015) and the fact that the Gauss-Southwell rule can be applied efficiently for coordinate descent in some forms of graphical models Nutini et al. (2015).

The BCMM algorithm of Hong et al. (2014) which use stochastic block coordinate updates inside of an ADMM formulation inspired our approach. But our algorithm perform multiple passes over all blocks before updating the multiplier; and we prove here stronger convergence rates.

We list related structured learning methods in Table 1 in Appendix B.5 with a comparison of their main characteristics.

Yen et al. (2016) is the most similar work to ours: the proposed algorithm constructs greedily an (initially sparse) working set of cliques, which is incremented at each epoch, while we perform stochastic updates on all cliques and possibly several passes over the data between each update of all Lagrange multipliers. Also, our work is leveraging the connection with SDCA, augmented Lagrangian methods and inexact gradient techniques, and we prove both linear convergence in the primal and the dual whereas they prove only linear convergence in the dual. Finally, our algorithm is outperforming other methods which is confirmed by our experiments.

3 CRF Learning

A discrete *conditional random field* (CRF) is a family of conditional distributions over a vector of discrete random variables $Y := (Y_1, \dots, Y_S)$ given the observation X. The form of CRF is assumed to be a product of *local functions* (a.k.a. *factors* or *clique functions*) that each depends on only a small number of random variables (i.e. a *clique*). If there exists multiple cliques take the same local function, then we can group cliques in terms of *clique types*. Specifically, let $w_\tau \in \mathbb{R}^{d_\tau}$ be the parameter vector associated with the clique type $\tau \in \mathcal{T}$, where $\mathcal{T}$ is the set of clique types. Let $\mathcal{C}$ denote the set of all cliques, and $\mathcal{C}_\tau$ denote the set of cliques of type $\tau \in \mathcal{T}$. Note that each clique c has a unique clique type, which we denote by τ_c. W.l.o.g., we consider only nodes $\mathcal{V}$ and edges $\mathcal{E}$, that is $\mathcal{C} = \mathcal{V} \cup \mathcal{E}$. But the framework can be generalized to higher-order cliques. We now introduce the density function of the CRF

$$p(y|x;w) := \frac{1}{Z(x,w)} \prod_{\tau\in\mathcal{T}} \prod_{c\in\mathcal{C}_\tau} \exp\Big(\langle w_\tau, \phi_c(x, y_c)\rangle\Big),$$

where $Z(x,w)$ is the *partition function*, and $\phi_c(x, y_c) \in \mathbb{R}^{d_{\tau_c}}$ is the *feature map* for clique c. Since all random variables are discrete, we use a one-hot vector $y_s \in \mathcal{Y}_s := \{u \in \{0,1\}^{k_s} \colon \|u\|_1 = 1\}$ to represent the value of Y_s. Here k_s is the cardinality of $\mathcal{Y}_s$. For a clique c, the value for the corresponding random variables is $y_c = \otimes_{s\in c}\, y_s \in \mathcal{Y}_c := \bigotimes_{s\in c} \mathcal{Y}_s$, where $\otimes$ (resp. $\bigotimes$) denotes the tensor product of vectors (resp. of spaces). Similarly, $y \in \mathcal{Y}$ is of the form $y = \otimes_{s\in\mathcal{V}}\, y_s$.

3.1 CRF as exponential family

Given a sample $(x^{(n)}, y^{(n)})$, for each clique c, let $\eta_c^{(n)}(w) := [\langle w_{\tau_c}, \phi_c(x^{(n)}, y_c)\rangle \colon y_c \in \mathcal{Y}_c]$; then $\eta^{(n)}(w) := [\eta_c^{(n)}(w) \colon c \in \mathcal{C}]$ is a *natural parameter* for the exponential family form of the conditional distribution $p(y \mid x^{(n)})$. The associated *sufficient statistics* is $T(y) := [y_c \colon c \in \mathcal{C}]$, and $\langle \eta^{(n)}(w), T(y)\rangle = \sum_c \langle \eta_c^{(n)}(w), y_c\rangle$. With these notations, $p(y \mid x^{(n)})$ has the exponential family form:

$$p\big(y \mid \eta^{(n)}(w)\big) = \exp\Big[\langle \eta^{(n)}(w), T(y)\rangle - F\big(\eta^{(n)}(w)\big)\Big],$$

where $F(u) := \log \sum_y \exp(\langle u, T(y)\rangle) = \log Z(x^{(n)}, w)$ is the log-partition function.

Given i.i.d. samples $\{(x^{(n)}, y^{(n)})\}_{1\le n\le N}$, the maximum likelihood estimator for w is computed by the maximizing $\sum_n \log p(y^{(n)} \mid x^{(n)}; w)$. Using the exponential family representation, we can rewrite this problem in two equivalent forms:

$$\max_w \sum_{n=1}^{N} \Big[\langle \eta^{(n)}(w), T(y^{(n)})\rangle - F\big(\eta^{(n)}(w)\big)\Big],$$

and $\min_w \sum_{n=1}^N F\big(\theta^{(n)}(w)\big)$, with $\theta^{(n)}(w)$ another natural parameter obtained via the affine transformation $\theta^{(n)}(w) = \eta^{(n)}(w) - \langle \eta^{(n)}(w), T(y^{(n)})\rangle \mathbf{1}$. Alternatively, by defining $\Psi^{(n)}$ as a sparse block matrix with $|\mathcal{T}| \times |\mathcal{C}|$ blocks, whose (τ_c, c)-th block is the matrix $\Psi_c^{(n)} \in \mathbb{R}^{d_{\tau_c} \times k_c}$ with

$$\Psi_c^{(n)} = [\phi_c(x^{(n)}, y_c) - \phi_c(x^{(n)}, y_c^{(n)}) \colon y_c \in \mathcal{Y}_c],$$

we can explicitly write $\theta_c^{(n)}(w) = \Psi_c^{(n)\mathsf{T}} w_{\tau_c}$ and $\theta^{(n)}(w) = \Psi^{(n)\mathsf{T}} w$.

W.l.o.g., we will assume $N = 1$ and drop the superscript (n) in the rest of the paper, since one may view N graphs as a single large graph with several connected components. The regularized maximum likelihood estimation with a regularization constant $\lambda > 0$ is thus formulated as

$$\min_w F\big(\theta(w)\big) + \frac{\lambda}{2}\|w\|_2^2. \tag{1}$$

In order to extend this formulation to cover as well max-margin learning (i.e., structured SVM), we consider the loss-augmented CRF learning introduced by Pletscher et al. (2010) and Hazan and Urtasun (2010), which leads to a slightly generalized formulation:

$$\min_w \gamma F\big(\tfrac{1}{\gamma}\theta_\ell(w)\big) + \frac{\lambda}{2}\|w\|_2^2, \tag{2}$$

where $\theta_\ell(w) := \theta(w) + \ell$ a new natural parameter, with $\ell = \big[[\ell_c(y_c^\star, y_c) \colon y_c \in \mathcal{Y}_c] \colon c \in \mathcal{C}\big]$ the user-defined loss and $\gamma \in (0, +\infty)$ the temperature hyperparameter. A detailed presentation of the loss-augmented CRF can be found in the Appendix A.

It is well known that the cost of gradient descent to optimize either (1) or (2) (for $\gamma > 0$) is prohibitive since $\nabla_{w_\tau} F(\theta(w)) = \sum_{c \in \mathcal{C}_\tau} \mathbb{E}_\theta[\Psi_c y_c]$ involves an expectation over the exponentially large space $\mathcal{Y}$. To exploit the underlying structure of the function F it is useful to working on the dual problem. Indeed, since F is convex, it has a variational representation based on conjugate duality:

$$F(\theta) = \max_\mu \langle \mu, \theta\rangle - F^*(\mu),$$

where F^* is the Fenchel conjugate of F, and the dual variable μ is called the *mean parameter*, since it is coupled with some natural parameter θ via the moment matching condition $\mu = \mathbb{E}_\theta[T(y)]$ as long as F and F^* are well defined. The set of valid mean parameters form the so called *marginal polytope* $\mathcal{M}$, which is defined as the convex hull of $\{T(y) : y \in \mathcal{Y}\}$. It is a classical result (Wainwright, 2008, Theorem 3.4) that

$$F^*(\mu) = -H_{\text{Shannon}}(\mu) + \iota_{\mathcal{M}}(\mu),$$

where $H_{\text{Shannon}}(\mu)$ denote the Shannon entropy of a CRF with mean parameter μ, and where $\iota_{\mathcal{M}}(\mu) = 0$ if $\mu \in \mathcal{M}$ and $\iota_{\mathcal{M}}(\mu) = +\infty$ otherwise.

4 Relaxed Formulations

In this section, we derive a general relaxed dual, primal and corresponding saddle-point formulations of the CRF learning problem, using first the classical local polytope relaxation, and then a relaxation of the marginalization constraints via an augmented Lagrangian. In addition, we propose a relaxation of the entropy, which is decomposable, and well defined even when the aforementioned constraints are relaxed. The resulting formulation is convex and is amenable to fast optimization algorithm that are presented in the following section.

4.1 Classical local polytope relaxation

Both $\mathcal{M}$ and $H_{\text{Shannon}}(\mu)$ are in general intractable due to the exponentially large structured-output space $\mathcal{Y}$ and they are typically replaced by convex surrogates.

It is common to relax $\mathcal{M}$ to a locally consistent counterpart– the *local consistency polytope* (Wainwright, 2008)

$$\mathcal{L} := \Big\{\mu \in \mathcal{I} \colon \sum_{y_t} \mu_{st}(y_s, y_t) = \mu_s(y_s), \forall st \in \mathcal{E}, \forall y_s\Big\},$$

where $\mathcal{I}$ denotes the Cartesian product of *simplex constraints* on each clique. Note that $\mathcal{L} \supseteq \mathcal{M}$, since any set of true marginals must satisfy the simplex constraints and the *marginalization constraints*, but not vice versa. Equivalently, if we define $A_s = I_{k_s} \otimes \mathbf{1}_{k_s}^\mathsf{T}$, the equality constraints can be written in a matrix form as $\mu_s - A_s \mu_{st} = 0$ for all $st \in \mathcal{E}$. Combining all equations, we have $A\mu = 0$, where A is a $|\mathcal{E}| \times |\mathcal{C}|$ block matrix (see Appendix F). So, we have equivalently $\mathcal{L} = \mathcal{I} \cap \{\mu \colon A\mu = 0\}$.

Since H_{Shannon} is also intractable for graphs with large tree-width, we will use an approximation H_{Approx} which will be constructed so as to be defined and concave on the whole set $\mathcal{I}$. We propose several entropy approximations suited to our needs in Section 4.3.

Definition 1. *Let $F_{\mathcal{I}}$ and $F_{\mathcal{L}}$ be the counterparts of F obtained by relaxing $\mathcal{M}$ to $\mathcal{I}$ and $\mathcal{L}$ respectively, which in other words the Fenchel conjugates of $F_{\mathcal{I}}^*$ and $F_{\mathcal{L}}^*$ when these are defined with H_{Approx}:*

$$F_{\mathcal{I}}(\theta_\ell) := \max_\mu \langle \mu, \theta_\ell\rangle - F_{\mathcal{I}}^*(\mu),$$
$$F_{\mathcal{L}}(\theta_\ell) := \max_\mu \langle \mu, \theta_\ell\rangle - F_{\mathcal{L}}^*(\mu),$$

with $F_{\mathcal{I}}^*(\mu) := -H_{\text{Approx}}(\mu) + \iota_{\mathcal{I}}(\mu)$ and $F_{\mathcal{L}}^*(\mu) := F_{\mathcal{I}}^*(\mu) + \iota_{\{A\mu=0\}}$.

Replacing F with $F_{\mathcal{L}}$ in (2) yields the relaxed primal

$$P(w) := \gamma F_{\mathcal{L}}\big(\tfrac{1}{\gamma}\theta_\ell(w)\big) + \frac{\lambda}{2}\|w\|_2^2. \tag{3}$$

The corresponding dual objective function is given by

$$D(\mu) := \langle \mu, \ell \rangle - \gamma F^*_{\mathcal{L}}(\mu) - \frac{1}{2\lambda}\|\Psi\mu\|_2^2. \tag{4}$$

See Appendix B.1 for a derivation.

4.2 A dual augmented Lagrangian

It is difficult to optimize $D(\mu)$, since the optimization requires some form of projection onto $\mathcal{L}$, which can be shown to be equivalent to perform graph-wise marginal inference (Collins et al., 2008). The difficulty is due to the coupling equality constraint $A\mu = 0$. Meshi et al. (2015b) proposed to relax $\iota_{\{A\mu=0\}}$ by a quadratic term $\frac{1}{2\rho}\|A\mu\|_2^2$, which corresponds to employ the penalty method (Bertsekas, 1982). They argue that it is not crucial to enforce exact $A\mu = 0$ in learning, since the relaxed problem works well in practice and enables an efficient optimization with only clique-wise updates. However, the penalty method is known as a fragile method due to the hyperparameter ρ. Unless we use a reasonably small ρ, or use a carefully designed scheduling to update ρ, the algorithm might be slow. On the other hand, using a large fixed value of ρ degrades the problem to independent logistic regression problems, it thereby leads to suboptimal solutions.

Instead, we propose to solve problem (4) as a saddle problem of the form $\max_\mu \min_\xi D_\rho(\mu, \xi)$ where D_ρ is the augmented Lagrangian

$$\begin{aligned} D_\rho(\mu, \xi) :=& \Big[\langle \ell, \mu \rangle - \gamma F^*_{\mathcal{I}}(\mu) + \langle \xi, A\mu \rangle\Big] \\ &- \Big[\frac{1}{2\rho}\|A\mu\|_2^2 + \frac{1}{2\lambda}\|\Psi\mu\|_2^2\Big], \end{aligned} \tag{5}$$

with ξ is the Lagrangian multiplier and $\rho > 0$.

Using duality again, we can derive a relaxed primal objective

$$\tilde{P}_\rho(w, \delta, \xi) := \gamma F_{\mathcal{I}}\Big(\frac{\theta_\ell(w) + A^\intercal\delta}{\gamma}\Big) + \frac{\lambda}{2}\|w\|_2^2 + \frac{\rho}{2}\|\delta - \xi\|_2^2,$$

so that $\min_{(w,\delta)} \tilde{P}_\rho(w, \delta, \xi)$ is a primal problem associated with the dual problem $\max_\mu D_\rho(\mu, \xi)$.

Strong duality between these two problems yields a representer theorem

$$w^\star = -\frac{1}{\lambda}\Psi\mu^\star, \quad \delta^\star = \xi^\star - \frac{1}{\rho}A\mu^\star \tag{6}$$

which provides a duality gap

$$\mathrm{gap}(w, \delta, \mu, \xi) := \tilde{P}_\rho(w, \delta, \delta) - D_\rho(\mu, \xi)$$

for the convergence of the maximization of $D_\rho(\mu, \xi)$ with respect to μ. Moreover, it is easy to check that $\min_{\xi,\delta} \tilde{P}_\rho(w, \delta, \xi) = P(w)$ because $\min_\delta F_{\mathcal{I}}(w + A^\intercal\delta) = F_{\mathcal{L}}(w)$ for any w. This shows that w^* defined above is also an optimum of the original primal problem $\min_w P(w)$. As a consequence, if a sequence μ^t converges to μ^* then the corresponding $w^t = -\frac{1}{\lambda}\Psi\mu^t$ converges to a solution of (2). For more details, see Appendix B.

4.3 Gini entropy surrogate

We seek a concave entropy surrogate H_{Approx} that decomposes additively on the cliques. Since the constraint $A\mu = 0$ is relaxed, we need a surrogate well defined on the whole set $\mathcal{I}$. The Bethe entropy (Yedidia et al., 2005) is generally non-concave. Its concave counterparts, such as the tree-reweighted entropy (Wainwright et al., 2005) or the region-based entropy (Yedidia et al., 2005; London et al., 2015), are only concave on the local consistency polytope, but non-concave[1] on $\mathcal{I} \setminus \mathcal{L}$ (i.e., when $A\mu \neq 0$).

Moreover, a generic difficulty with entropies, is that $\log(\cdot)$ does not have Lipschitz gradients, which prevents the direct application of proximal methods with usual quadratic proximity terms. We thus propose a coarse but convenient entropy surrogate of the form:

$$H_{\mathrm{Approx}}(\mu) = H_{\mathrm{Gini}}(\mu) := \sum_{c \in \mathcal{C}} (1 - \|\mu_c\|_2^2).$$

A more precise choice could be the second-order Taylor approximation of the *oriented tree-reweighted entropy* (Globerson and Jaakkola, 2007) around the uniform distribution (denoted by $H_{\mathrm{GTRW}}(\mu)$), which also meet our requirements, since it is concave (although not strongly concave) in $\mathcal{I}$ and smooth. However, we do not find a significant improvement in practice by using this surrogate. More details about $H_{\mathrm{GTRW}}(\mu)$ can be found in Appendix C.

5 Algorithm

Given the form of the entropy surrogate proposed, $\mu \mapsto D_\rho(\mu, \xi)$ decomposes as a sum of (strongly) convex separable terms over the block associated to cliques plus a smooth term. It can be solved efficiently by block-coordinate proximal schemes, such as the proximal stochastic dual coordinate descent (SDCA) (Shalev-Shwartz and Zhang, 2014), which has both guarantees of linear convergence in the primal and the dual. See Appendix E for the detailed form of $D_\rho(\mu, \xi)$.

[1] The Bethe entropy and its concave variants admit the general form $H_{\mathrm{Bethe}}(\mu) = \sum_{s \in \mathcal{V}} c_s H_s(\mu_s) + \sum_{st \in \mathcal{E}} c_{st} H_{st}(\mu_{st})$, where c_s and c_{st} are counting numbers. Even when H_{Bethe} is concave on $\mathcal{L}$, some of the counting numbers can be negative.

Algorithm 1 IDAL Method

Input: $T_{\text{in}}, T_{\text{ex}}, \epsilon$
Initialize: $\hat{\mu}_c^0 = \frac{1}{k_c}\mathbf{1}$ for all $c \in \mathcal{C}$ and $\xi^0 = 0$
for $t = 1, \ldots, T_{\text{ex}}$ **do**
  $\xi^t = \xi^{t-1} - \frac{1}{L} A\hat{\mu}^{t-1}$; $\mu^{t,0} = \hat{\mu}^{t-1}$
  for $s = 1, \ldots, T_{\text{in}}$ **do**
    Draw a clique c uniformly at random
    $u_c = \texttt{prox\_block\_update}(c, \mu^{t,s-1})$
    $\mu_c^{t,s} = u_c$; $\mu_{-c}^{t,s} = \mu_{-c}^{t,s-1}$; $\hat{\mu}^t = \mu^{t,s}$ if $s = T_{\text{in}}$
  end for
  Stop if $G_t \leq \epsilon$ and $\|A\hat{\mu}^t\|^2 \leq \epsilon$
end for
Output: $\hat{\mu}^t, \xi^t$

So, a natural approach to solve $\min_\xi \max_\mu D_\rho(\mu, \xi)$ is to consider ADMM type schemes that update the parameters μ_c one after the others and ξ periodically as well. Since naive forms of multi-block ADMM that perform cyclic updates are not necessarily convergent (Chen et al., 2016), we consider an algorithm similar to the block coordinate method of multipliers (BCMM) by Hong et al. (2014) that randomizes the order of the blocks: we propose to perform dual stochastic block coordinate ascent (SDCA) on the variables μ_c to partially maximize $D_\rho(\mu, \xi)$ w.r.t. μ and to perform regularly a gradient descent step w.r.t. ξ. Our algorithm, referred as inexact dual augmented Lagrangian (IDAL) method, can be interpreted as an inexact gradient descent algorithm on the function $\xi \mapsto d(\xi) := \max_\mu D(\mu, \xi)$.

To be precise, assume that at epoch t, ξ takes the value ξ^t and $\hat{\mu}^{t-1}$ is the current value obtained from the previous epoch. The proposed algorithm (Algorithm 1) takes T_{in} stochastic block-coordinate proximal steps on μ (i.e. $\mu_c^{\text{new}} = \texttt{prox\_block\_update}(c, \mu)$), then denoting $\hat{\mu}^t$ the obtained value at the end of this epoch, since $A\hat{\mu}^t$ is an approximate gradient of $d(\xi^t)$, ξ is updated with $\xi^{t+1} = \xi^t - \frac{1}{L} A\hat{\mu}^t$, where L is the Lipschitz constant of $d(\xi)$.

To detect the convergence of the algorithm, we use the practical criteria $G_t := \text{gap}(w(\hat{\mu}^t), \delta(\hat{\mu}^t, \xi^t), \hat{\mu}^t, \xi^t) \leq \epsilon$ and $\|A\hat{\mu}^t\|^2 \leq \epsilon$, where $w(\hat{\mu}^t), \delta(\hat{\mu}^t, \xi^t)$ are defined via the representer theorem (see Appendix B.4). The former measures the suboptimality of D_ρ in μ, while the latter measures the suboptimality of $d(\xi)$.

6 Convergence Analysis

In this section, we study the convergence rate of our algorithm. First, we show that if we use an iterative and linearly convergent algorithm $\mathcal{A}$ to approximately solve $\min_\mu D(\mu, \xi)$, and if we use warm starts, that is, following the notations of the previous section, we use $\hat{\mu}^{t-1}$ as the initial value to solve $\min_\mu D(\mu, \xi^t)$, then running $\mathcal{A}$ for a fixed number of iterations is sufficient to guarantee global linear convergence in the primal and in the dual. We show that SDCA or simple block-coordinate proximal gradient descent are applicable as the algorithm $\mathcal{A}$.

6.1 Conditions for global linear convergence

To study the convergence of the algorithm, we consider the following quantities:

- $\bar{\mu}^t := \mu^*(\xi^t) = \text{argmin}_\mu D_\rho(\mu, \xi^t)$.
- $\hat{\mu}^t := \mu^{t, T_{\text{in}}}$, the solution obtained after running T_{in} inner iterations of the algorithm $\mathcal{A}$ given ξ^t.
- The exact gradient $g_t := \nabla d(\xi^t) = A\mu^*(\xi^t)$ and its approximate version $\hat{g}_t := A\hat{\mu}^t$.
- D_ρ-suboptimality: $\Delta_t^s := D_\rho(\bar{\mu}^t, \xi^t) - D_\rho(\mu^{t,s}, \xi^t)$, and its value at the end of each epoch $\hat{\Delta}_t = \Delta_t^{T_{\text{in}}} = \Delta_{t+1}^0$.
- d-suboptimality: $\Gamma_t := d(\xi^t) - d(\xi^*)$.

Theorem 1 (Linear convergence of the outer iteration)**.** *Suppose we have an algorithm $\mathcal{A}$ to approximately solve $\max_\mu D_\rho(\mu, \xi^t)$ such that $\mathbb{E}[\hat{\Delta}_t] \leq (1-\pi)^{T_{\text{in}}} \mathbb{E}[\Delta_t^0]$, where $\pi \in (0, 1)$ is the condition number of $D_\rho(\mu, \cdot)$. There exists $0 < \kappa < 1$, such that for any $\beta \in (0, 1)$, it is sufficient to run $T_{in} \geq \frac{\log(\beta)}{\log(1-\pi)}$ inner iterations on μ to guarantee that, after T_{ex} gradient steps on ξ, the suboptimalities $\hat{\Delta}_{T_{\text{ex}}}$ and $\Gamma_{T_{\text{ex}}}$ are bounded from above:*

$$\left\| \begin{matrix} \mathbb{E}[\hat{\Delta}_{T_{\text{ex}}}] \\ \mathbb{E}[\Gamma_{T_{\text{ex}}}] \end{matrix} \right\| \leq \lambda_{\max}(\beta)^{T_{\text{ex}}} C \left\| \begin{matrix} \mathbb{E}[\hat{\Delta}_0] \\ \mathbb{E}[\Gamma_0] \end{matrix} \right\|,$$

where $\lambda_{\max}(\beta)$ is the largest eigenvalue of the matrix

$$M(\beta) = \begin{bmatrix} 6\beta & 3\beta \\ 1 & 1-\kappa \end{bmatrix}.$$

So, if β is chosen so that $\lambda_{\max}(\beta) < 1$, the algorithm is linearly convergent.

The constant κ in the Theorem is of the form $\kappa = \frac{\tau}{L}$ with L the Lipschitz constant of d and τ is essentially a restricted strong convexity constant for d obtained by Hong and Luo (2012) (see Lemma 2 in Appendix D.2). κ could therefore be interpreted as a pseudo-condition number for d.

Corollary 1. *To ensure that $\mathbb{E}\hat{\Delta}_t \leq \epsilon$ and $\mathbb{E}\Gamma_t \leq \epsilon$ it is enough to run the algorithm for a total number of inner iteration $T_{\text{tot}} := T_{\text{in}} T_{\text{ex}}$ such that*

$$T_{\text{tot}} \geq \frac{\log(\beta)}{\log \lambda_{\max}(\beta) \log(1-\pi)} \log(\epsilon).$$

To reason in terms of rate, if the rate of convergence is r then we would have $T_{\text{tot}} \geq \frac{\log(\epsilon)}{\log(1-r)}$. So identifying the rate of convergence of the algorithm yields $r = 1 - \exp\left(\frac{\log(1-\pi)\log(\lambda_{\max}(\beta))}{\log(\beta)}\right)$. In order to provide an interpretation of this expression, we make some assumption to simplify the expression.

In particular, we show in Appendix D.4 that to have $\lambda_{\max}(\beta) < 1$ we should have $\beta = \alpha\kappa$ with $\alpha < \frac{1}{3(1+2\kappa)}$. So in particular $\alpha < \frac{1}{9}$ is sufficiently small. When $\kappa < \frac{1}{2}$ and $\alpha < \frac{1}{6}$, we also show that $\lambda_{\max}(\alpha\kappa) \leq 1 - \kappa + 6\alpha\kappa$. Setting $\alpha = \frac{1}{12}$, we get

$$r \geq 1 - (1-\pi)^{\frac{\log(1-\frac{\kappa}{2})}{\log(\frac{\kappa}{12})}} \geq \frac{\log(1-\frac{\kappa}{2})}{\log(\frac{\kappa}{12})}\,\pi \geq \frac{\kappa}{-2\log(\frac{\kappa}{12})}\,\pi,$$

and, for κ and π small, r is of order $\frac{\kappa}{-\log(\kappa)}\pi$, which up to the log factor is a multplicative combination of the (pseudo)-condition numbers of the inner and outer problems.

6.2 Convergence results with SDCA

As announced, given that for fixed ξ, $\min_\mu D_\rho(\mu, \xi)$ is a composite problem in μ with a smooth part and a strongly concave part, a good candidate for $\mathcal{A}$ is the proximal stochastic dual coordinate descent (SDCA) algorithm proposed by Shalev-Shwartz and Zhang (2016).

Proposition 1. *If $\mathcal{A}$ is SDCA, denote by $|\mathcal{C}|$ the total number of cliques, σ_c the strong convexity constant of $\mu_c \mapsto -H_{\text{Approx}}(\mu)$, and L_c the Lipschitz constant of $\mu_c \mapsto \frac{1}{\lambda}\Psi^\intercal w(\mu) + \frac{1}{\rho}A^\intercal\delta(\mu, \xi^t)$, then we have $\pi = \min_{c\in\mathcal{C}} \frac{\sigma_c}{|\mathcal{C}|(\sigma_c + L_c)}$ such that $\mathbb{E}_c[\hat{\Delta}_t] \leq (1-\pi)^{T_{in}}\mathbb{E}_c[\Delta_t^0]$.*

Note that there exists other algorithms that are applicable as $\mathcal{A}$. In fact, maximizing exactly D_ρ w.r.t. μ_c does not have a closed-form solution. We have to use a line search variant of SDCA for `prox_block_update`. Alternatively, we can view $D_\rho(\mu, \xi)$ as

$$-D_\rho(\mu, \xi) = \tilde{D}_\rho(\mu, \xi) + \sum_{c\in\mathcal{C}} \iota_\Delta(\mu_c).$$

Note that $\tilde{D}_\rho(\mu, \xi)$ is smooth and strongly convex in μ. Thus, we can use a block-coordinate proximal gradient method (Karimi et al., 2016) to approximately maximize $D_\rho(\mu, \xi)$ w.r.t. μ, although this algorithm gives a worse linear rate than SDCA. We show in the following proposition that it is indeed a valid candidate for $\mathcal{A}$.

Proposition 2. *Denote by $\tilde{L}$ and $\tilde{\sigma}$ the smoothness and strong convexity constants of $\tilde{D}_\rho$ respectively. Let U_c be the matrix such that $\mu_c = U_c^\intercal \mu$. If $\mathcal{A}$ is constructed by setting* `prox_block_update` *as*

$$\mu_c^{t,s+1} = \arg\min_{u\in\Delta} \left[\langle U_c^\intercal \nabla \tilde{D}_\rho(\mu, \xi), u\rangle + \frac{\tilde{L}}{2}\|u - \mu_c^{t,s}\|^2\right],$$

where c is selected uniformly at random, then we have $\pi = \frac{\tilde{\sigma}}{\tilde{L}|\mathcal{C}|}$, such that $\mathbb{E}_c[\hat{\Delta}_t] \leq (1-\pi)^{T_{in}}\mathbb{E}_c[\Delta_t^0]$.

However, SDCA enables us to bound the duality gap by the increase of D_ρ, which leads to a guarantee of linear convergence in the primal.

Corollary 2. *Let $\hat{w}^t = w(\hat{\mu}^t)$. If $\mathcal{A}$ is SDCA, then*

$$\mathbb{E}[P(\hat{w}^t) - P(w^\star)] \leq \frac{1}{\pi}\mathbb{E}[\hat{\Delta}_t] + \mathbb{E}[\Gamma_t].$$

Hence, if $\mathbb{E}[\hat{\Delta}_t + \Gamma_t]$ converges to 0 linearly, then so does $\mathbb{E}[P(\hat{w}^t) - P(w^\star)]$.

Optimization with inexact gradients (Devolder et al., 2014) and inexact proximal operators (Schmidt et al., 2011) have been shown be able to yield the same convergence rate as its exact counterparts, provided that errors decrease at a certain rate. The strategy of IDAL is somehow similar to these ideas except that we quantify directly the gradient approximation in terms of T_{in}, and show that thanks to warm-starts T_{in} can be constant during optimization. The use of warm-start is similar to its use in the meta-algorithm proposed by Lin et al. (2016), who use inexact gradient descent on the Moreau-Yosida regularization of a non-smooth objective. In our context, this approach would actually be applicable by working on $P_\rho(w, \xi)$ instead of working in the dual. An investigation in this direction is of interest but beyond the scope of this paper.

7 Experiments

We evaluate our algorithm IDAL on three different CRF models including 1) a simulated Gaussian mixture Potts model with grid graph and two clique types (nodes and edges); 2) a semantic segmentation model with planar graph and two clique types (nodes and edges); 3) a multi-label classification model with fully-connected graph and unique clique type for all cliques.

We compare with algorithms using only clique-wise oracles for solving $\min_\xi \max_\mu D_\rho(\mu, \xi)$, namely, the soft-constrained block-coordinate Frank-Wolfe algorithm (SoftBCFW) by Meshi et al. (2015b) and the greedy direction method of multipliers (GDMM) algorithm by Yen et al. (2016). Note that SoftBCFW in fact solves only the special case $\max_\mu D_\rho(\mu, \xi \equiv 0)$, thus it will converge to a different point than IDAL. In addition, we include a third baseline for the special case using SDCA (referred as SoftSDCA). Since SoftBCFW and GDMM have been shown outperforming other baselines such as Lacoste-Julien et al. (2013), Meshi et al. (2010) and Hazan and Urtasun (2010), we will not make an extensive comparison for all these algorithms.

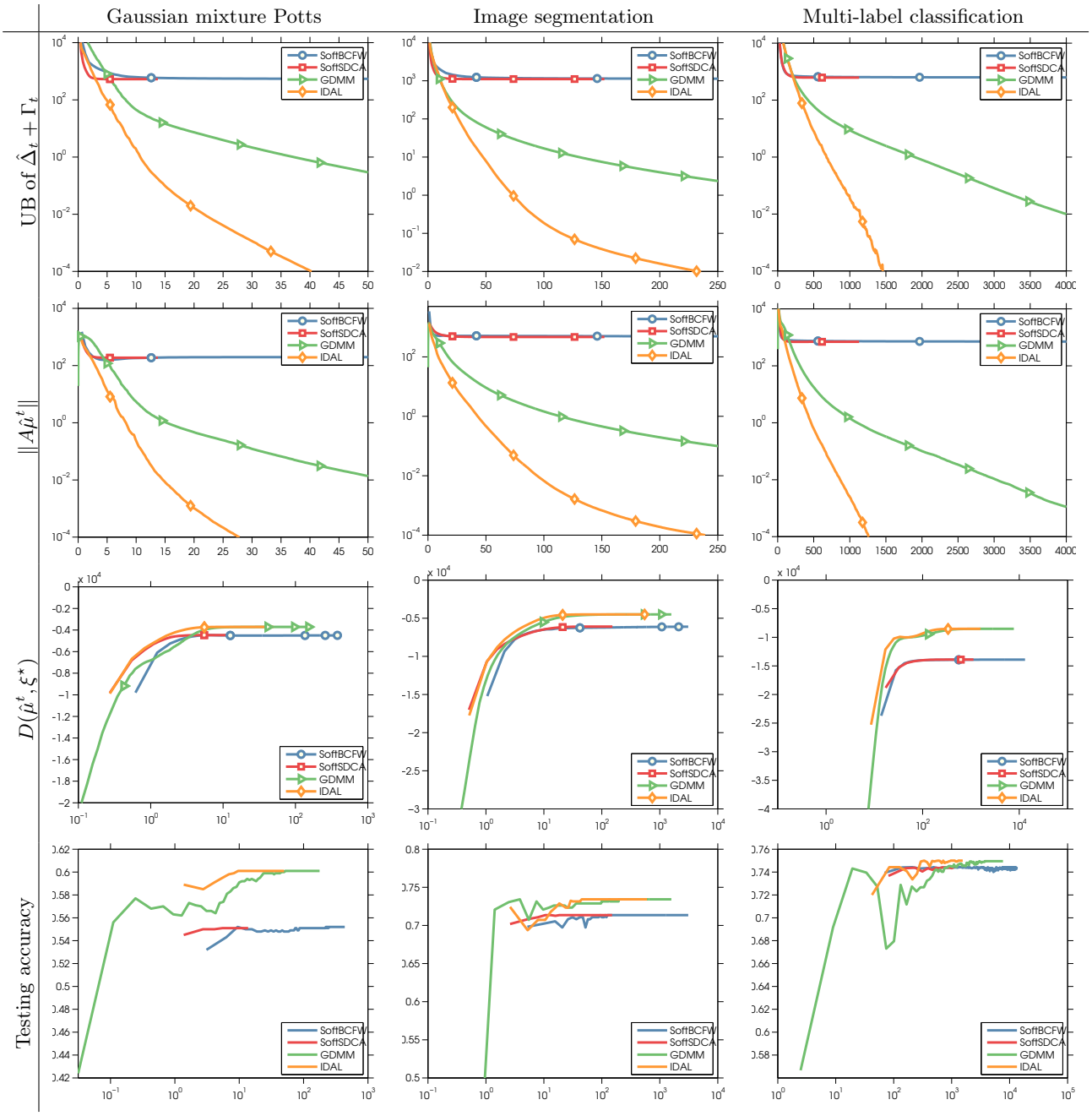

Figure 1: The comparison between IDAL and other baselines. For the best choices of λ and γ, we set $(\lambda = 10.0, \gamma = 10.0)$ for Gaussian mixture Potts, $(\lambda = 10.0, \gamma = 1.0)$ for multi-label classification and $(\lambda = 10.0, \gamma = 0.001)$ for semantic segmentation. Note that x-axis denote time (second).

7.1 Setup

Gaussian mixture Potts models This is an extension of the Potts model given observations, whose conditional density function is defined via Bayes' rule $p(y|x) \propto p(x|y)p(y)$, with $p(y)$ a Potts distribution associated with a grid graph and parameterized by $w_{\text{binary}} \in \mathbb{R}^{k^2}$, and with $p(x|y) = \prod_s p(x_s|y_s)$ assumed to factorize into independent conditional Gaussian distributions with canonical parameters $w_{\text{unary}} \in \mathbb{R}^{2k}$, i.e., $p(x_s|y_s) \propto \exp(\langle w_{\text{unary}}(y_s), [x_s, x_s^2] \rangle)$. We consider a 10×10 grid graph with node cardinality $k = 5$. To generate the data, we first draw the label y from $p(y)$, and then the observation x_s is generated from the conditional Gaussian $p(x_s|y_s)$ for each node. The simulated dataset contains 100 samples and is equally divided for training and testing.

Semantic image segmentation We consider a typical CRF model used in computer vision for labeling image pixels with semantic classes. The graph is built upon clustering pixels into superpixels. Each superpixel defines a node. Two superpixels with a shared boundary define an edge. The CRF model takes the form $p(y|x) \propto \exp(\sum_s w_{\text{unary}}^\intercal \psi_s(x, y_s) + \sum_{s,t} w_{\text{binary}}^\intercal \psi_{st}(x, y_s, y_t))$, where $\psi_s(x, y_s)$ measure the intra-cluster compatibility within the superpixel s, and $\psi_{st}(x, y_s, y_t)$ measure the inter-cluster compatibility between superpixels s and t. We conduct the experiment on the *MSRC-21* dataset introduced by Shotton et al. (2006), which has 21 classes, 335 training images and 256 testing images.

Multi-label classification The task for this problem is assigning each input vector a set of binary target labels. It is natural to model the inter-label dependencies by CRFs that treat each label as a node in a fully connected label graph. Following Finley and Joachims (2008), we define the CRF density function as $p(y|x) \propto \exp(\sum_s w_s^\intercal \phi_s(x, y_s) + \sum_{s,t} w_{st}^\intercal \phi_{st}(y_s, y_t))$, where the feature maps are specified as $\phi_s(x, y_s) = y_s \otimes x$ for each node and $\phi_{st}(y_s, y_t) = y_s \otimes y_t$ for each edge. We conduct the experiments on the *Yeast* dataset[2], which contains 1500 training samples and 917 testing samples. Each sample has 14 labels and 103 attributes.

Hyperparameters In theory, T_{in} could be very large depending on the choice of α and the condition number. We find that in practice only a relatively small T_{in} is needed. We empirically choose $T_{\text{in}} = 2|\mathcal{C}|$. We set the number of outer iterations $T_{\text{ex}} = 3000$ and the stopping threshold $\epsilon = 10^{-6}$. The ranges of λ is pre-defined as $\{10, 1.0, 0.01, 0.001\}$ and the range of γ is $\{100.0, 10.0, 1.0, 0.001\}$. For each experiment, we choose the best λ and γ in terms of the validation accuracy and a reasonable running time (not all experiments finished in 3000 outer iterations). We set $\rho = 1.0$, as it is the value works pretty well for all algorithms.

7.2 Results

To compare IDAL with GDMM, we use the criterion $P_\rho(\hat{w}^t, \hat{\delta}^t, \xi^t) - D_\rho(\hat{\mu}^t, \xi^t) + P_\rho(\hat{w}^t, \hat{\delta}^t, \xi^t) - D_\rho(\bar{\mu}^{T_{\text{ex}}}, \xi^{T_{\text{ex}}})$, which is an upper bound of the theoretical quantity $\hat{\Delta}_t + \Gamma_t$ that we analyzed. To compare IDAL with SoftBCFW, since $\xi = 0$ for SoftBCFW, we use the criterion $D_\rho(\hat{\mu}^t, \xi^\star)$, in which $\xi^\star$ is obtained from running IDAL to convergence. Besides, we also use the criteria $\|A\hat{\mu}^t\|^2$ (it measures the convergence of $d(\xi)$, since $\nabla d(\xi^t) \simeq A\hat{\mu}^t$) and the testing accuracy, which are applicable for all three algorithms. The results are shown in Figure 1.

There are several interesting points that we can say based on the results. 1) By tightening the marginalization constraints $A\mu = 0$, it does help to gain a better testing accuracy (IDAL and GDMM gain $\sim 3\%$ over SoftBCFW); 2) Based on the curves of $D_\rho(\mu, \xi^*)$, we can see that it is key to approach μ^* by first obtaining ξ^*, which again shows the importance of enforcing exactness of the local consistency polytope; 3) IDAL is shown to be a faster algorithm than GDMM. One possible reason is that GDMM is in fact an active-set algorithm, which means the number of updated cliques at very beginning is insufficient comparing to IDAL. Based on our analysis, we have shown that the quality of the approximate gradient $\hat{g}_t$ depends on T_{in}. Therefore, it is very likely that GDMM suffers from a slow convergence because of the poor gradients.

8 Conclusion

We proposed a relaxed dual augmented Lagrangian formulation for CRF learning, in which, thanks to dual decomposition, SDCA can be used to partially optimize over mean parameters in order to yield sufficiently approximate multiplier gradients. Our theoretical analysis shows that if warm-starts are leveraged and multiplier gradients are approximated with a linearly convergent algorithm, global linear convergence can be obtained. If SDCA is used, linear convergence is obtained both in the primal and for the convergence of the dual Lagrangian method.

Comparing to other baselines such as GDMM and SoftBCFW, our algorithm is faster in terms of the distance to the optimal objective function value (i.e. $\hat{\Delta}_t + \Gamma_t$) and the feasibility of the constraints $\|A\mu\|_2^2$.

It would be of interest to investigate the use of the same dual augmented Lagrangian formulation for both inference and learning, since according to Wainwright (2006) this should improve the performance.

In future work, we intend to investigate applications to other problems in machine learning, the use of Nesterov acceleration or quasi-Newton methods for multiplier updates, or the connection to other approaches based on Moreau-Yosida regularization.

References

Bertsekas, D. P. (1982). The method of multipliers for equality constraints. In *Constrained optimization and Lagrange Multiplier methods*. Athena scientific.

Chen, C., He, B., Ye, Y., and Yuan, X. (2016). The direct extension of admm for multi-block convex

[2] `http://sourceforge.net/projects/mulan/files/datasets/yeast.rar`

minimization problems is not necessarily convergent. *Mathematical Programming*, 155(1-2):57–79.

Collins, M., Globerson, A., Koo, T., Carreras, X., and Bartlett, P. L. (2008). Exponentiated gradient algorithms for conditional random fields and max-margin markov networks. *Journal of Machine Learning Research*, 9:1775–1822.

Defazio, A., Bach, F., and Lacoste-Julien, S. (2014). SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems*, pages 1646–1654.

Devolder, O., Glineur, F., and Nesterov, Y. (2014). First-order methods of smooth convex optimization with inexact oracle. *Mathematical Programming*, 146(1-2):37–75.

Finley, T. and Joachims, T. (2008). Training structural SVMs when exact inference is intractable. In *International Conference on Machine Learning (ICML)*, pages 304–311.

Globerson, A. and Jaakkola, T. (2007). Convergent propagation algorithms via oriented trees. In *24th Conference on Uncertainty in Artificial Intelligence*.

Hazan, T. and Urtasun, R. (2010). A primal-dual message-passing algorithm for approximated large scale structured prediction. In *NIPS*.

Hong, M., Chang, T.-H., Wang, X., Razaviyayn, M., Ma, S., and Luo, Z.-Q. (2014). A block successive upper bound minimization method of multipliers for linearly constrained convex optimization. *arXiv preprint arXiv:1401.7079*.

Hong, M. and Luo, Z.-Q. (2012). On the linear convergence of the alternating direction method of multipliers. *Mathematical Programming*, pages 1–35.

Karimi, H., Nutini, J., and Schmidt, M. (2016). Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 795–811. Springer.

Komodakis, N. (2011). Efficient training for pairwise or higher order CRFs via dual decomposition. In *CVPR*.

Komodakis, N., Paragios, N., and Tziritas, G. (2007). Mrf optimization via dual decomposition: Message-passing revisited. In *ICCV*.

Kulesza, A. and Pereira, F. (2007). Structured learning with approximate inference. In *Advances in neural information processing systems*, pages 785–792.

Lacoste-Julien, S., Jaggi, M., Schmidt, M., and Pletscher, P. (2013). Block-coordinate frank-wolfe optimization for structural SVMs. In *ICML*.

Lin, H., Mairal, J., and Harchaoui, Z. (2016). Quickening: A generic quasi-newton algorithm for faster gradient-based optimization. *arXiv preprint arXiv:1610.00960*.

London, B., Huang, B., and Getoor, L. (2015). The benefits of learning with strongly convex approximate inference. In *ICML*.

Martins, A. F. T., Figueiredo, M. A. T., Aguiar, P. M. Q., Smith, N. A., and Xing, E. P. (2015). Ad3: Alternating directions dual decomposition for map inference in graphical models. *JMLR*.

Meshi, O., Mahdavi, M., and Schwing, A. G. (2015a). Smooth and strong: MAP inference with linear convergence. In *NIPS*.

Meshi, O., Sontag, D., Globerson, A., and Jaakkola, T. S. (2010). Learning efficiently with approximate inference via dual losses. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 783–790.

Meshi, O., Srebro, N., and Hazan, T. (2015b). Efficient training of structured SVMs via soft constraints. In *AISTATS*.

Nutini, J., Schmidt, M., Laradji, I. H., Friedlander, M., and Koepke, H. (2015). Coordinate descent converges faster with the Gauss-Southwell rule than random selection. *arXiv preprint arXiv:1506.00552*.

Pletscher, P., Ong, C. S., and Buhmann, J. M. (2010). Entropy and margin maximization for structured output learning. In *ECML*.

Savchynskyy, B., Kappes, J., Schmidt, S., and Schnörr, C. (2011). A study of nesterov's scheme for lagrangian decomposition and map labeling. In *CVPR*.

Schmidt, M., Babanezhad, R., Ahmed, M. O., Defazio, A., Clifton, A., and Sarkar, A. (2015). Non-uniform stochastic average gradient method for training conditional random fields. *arXiv preprint arXiv:1504.04406*.

Schmidt, M., Roux, N. L., and Bach, F. (2013). Minimizing finite sums with the stochastic average gradient. *arXiv preprint arXiv:1309.2388*.

Schmidt, M., Roux, N. L., and Bach, F. R. (2011). Convergence rates of inexact proximal-gradient methods for convex optimization. In *Advances in neural information processing systems*, pages 1458–1466.

Shalev-Shwartz, S. and Zhang, T. (2014). Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. In *ICML*.

Shalev-Shwartz, S. and Zhang, T. (2016). Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. *Mathematical Programming*, 155(1-2):105–145.

Shotton, J., Winn, J., Rother, C., and Criminisi, A. (2006). Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *European conference on computer vision*, pages 1–15. Springer.

Sontag, D., Meltzer, T., Globerson, A., Jaakkola, T., and Weiss, Y. (2008). Tightening lp relaxations for map using message passing. In *24th Conference on Uncertainty in Artificial Intelligence*.

Tang, K., Ruozzi, N., Belanger, D., and Jebara, T. (2016). Bethe learning of graphical models via map decoding. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 1096–1104.

Wainwright, M. J. (2006). Estimating the wrong graphical model: Benefits in the computation-limited setting. *JMLR*.

Wainwright, M. J. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*.

Wainwright, M. J., Jaakkola, T. S., and Willsky, A. S. (2005). A new class of upper bounds on the log partition function. *IEEE Transactions on Information Theory*, 51(7):2313–2335.

Yedidia, J. S., Freeman, W. T., and Weiss, Y. (2005). Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Transactions on information theory*, 51(7):2282–2312.

Yen, I. E.-H., Huang, X., Zhong, K., Zhang, R., Ravikumar, P. K., and Dhillon, I. S. (2016). Dual decomposed learning with factorwise oracle for structural svm of large output domain. In *Advances in Neural Information Processing Systems*, pages 5024–5032.