Convex Relaxations for Markov Random Field MAP estimation

Timothee Cour

GRASP Lab, University of Pennsylvania

September 2008

Abstract

Markov Random Fields (MRF) are commonly used in computer vision and maching learning applications to model interactions of interdependant variables. Finding the Maximum Aposteriori (MAP) solution of an MRF is in general intractable, and one has to resort to approximate solutions. We review some of the recent literature on convex relaxations for MAP estimation. Our starting point is to notice the MAP estimation (a discrete problem) is in fact equivalent to a real-valued but non-convex Quadratic Program (QP). We reformulate some of those relaxations and see that we can distinguish two main strategies:
1) optimize a convex upper-bound of the (non-convex) cost function (L2QP, CQP, our spectral relaxation);
2) reformulate as a linear objective using lift-and-project and optimize over a convex upper-bound of the (non-convex) feasible set (SDP, SOCP, LP relaxations).

We analyse these relaxations according to the following criteria: optimality conditions, relative dominance relationships, multiplicative/additive bounds on the quality of the approximation, ability to handle arbitrary clique size, space/time complexity and convergence guarantees. We will show a few surprising results, such as the equivalence between the CQP relaxation (a quadratic program) and the SOCP relaxation (containing a linear objective), and furthermore show that a large set of SOCP constraints are implied by the local marginalization constraint.

Along the way, we also contribute a few new results. The first one is a $\frac{1}{k^{c-1}}$ multiplicative approximation bound for an MRF with *arbitrary* clique size c and k labels, in the general case (extending the pairwise case $c = 2$). The second one is a tighter additive bound for CQP and LP relaxation in the general case (with $k = 2$ labels), that also has the big advantage of being invariant to reparameterizations. The new bound involves a *modularity norm* instead of an ℓ_1 norm. We also show that a multiplicative bound δ for the LP relaxation would imply $\delta \leq \frac{1}{2}$ (for $k = 2$), putting LP on par with other convex relaxations such as L2QP. Finally we characterize the equivalence classes of a (broader) class of reparameterizations, show their dimension, and how a basis can be used to generate potentially tigher relaxations. We believe those contributions are novel.

Contents

Chapter 1

Introduction

A number of problems in Computer Vision and Machine Learning can be formulated in a probabilistic setting using Markov Random Fields (MRF). Classical examples include stereo vision, image restoration, image labeling and graph matching. In each case, a set of interdependent variables can be assigned a range of labels, with a probability attached to each joint assignment. Inference in such a graphical model consists in finding the configuration with maximum a posteriori probability (MAP). In general, the inference problem is intractable, but there are interesting cases where it can be solved in polynomial time, such as tree-structured MRF (or with bounded tree-width), with a single cycle and binary variables, with convex priors[1], or binary MRF with submodular clique potentials[2].

MRFs have been studied extensively since the 1970's, and a lot of work has been focused on developping approximation algorithms for the MAP problem. Bayesian methods such as Belief Propagation (BP)[3, 4], Generalized BP and Tree Reweighted BP[5] are optimal in trees as well as certain graphs with cycles. In the general case, when the max-product version of BP converges, the assignment is guaranteed to be locally optimal in a large neighborhood[4]. However, there is no general convergence guarantee and BP may fail to converge even in simple graphs. Energy Minimization methods such as Graph Cuts[6, 2] have been successfully applied to early vision applications, often on planar graphs with nearest neighbor connectivity. For binary MRFs with submodular clique potentials, Graph Cuts are provably optimal. For multiple label MRFs, [6] introduces $\alpha - \beta$ swaps and α expansion moves that find solutions which are locally optimal with respect to large moves, but with some restrictions on the clique potentials.

We review in this report some of the recent literature on convex relaxations for MAP estimation, including L2QP[7, 8], CQP[9], our spectral relaxation[10], SDP[11], SOCP[12, 13], and LP[14, 15] relaxations. We start by showing that MAP estimation (a discrete problem) is in fact equivalent to a real-valued but non-convex Quadratic Program (QP). We reformulate some of these relaxations and see that two main strategies emerge:
1) optimize a convex upper-bound of the (non-convex) cost function (L2QP, CQP, our spectral relaxation);
2) reformulate as a linear objective using lift-and-project and optimize over a convex upper-bound of the (non-convex) feasible set (SDP, SOCP, LP relaxations).

We will encounter a few recurring themes underlying those relaxations. One such theme is the convexity of the MAP value in terms of MRF parameters, which forms the basis for decomposition techniques such as TRW and Dual Decomposition for MRF. We illustrate in fact a parallel between tree decomposition techniques for max-marginals/MAP estimation, and the planar graph decomposition for estimating sum-marginals and the partition function. A second recurring theme concerns reparameterizations, which encode equivalence classes of MRF parameters. One can view message-passing updates alternatively as reparameterizations, or as fixed-point updates for the dual variables of the LP relaxation. We reformulate certain convex relaxations (CQP, L2QP, spectral relaxation) as seeking a best reparameterization.

We analyse the relaxations mentioned above according to the following criteria: optimality conditions, relative dominance relationships, multiplicative/additive bounds on the quality of the approximation, ability to handle arbitrary clique size, space/time complexity and convergence guarantees. We will show a few surprising results, such as the equivalence between the CQP relaxation (a quadratic program) and the SOCP relaxation (containing a linear objective), and furthermore show that a large set of SOCP constraints are implied by the local marginalization constraint.

Along the way, we also contribute a few new results. The first one is a $\frac{1}{k^{c-1}}$ multiplicative approximation bound for an MRF with *arbitrary* clique size c and k labels, in the general case. The second one is a tighter additive bound for CQP and LP relaxation in the general case (with $k = 2$ labels), that also has the big advantage of being invariant to reparameterizations. The new bound involves a *modularity norm* instead of an ℓ_1 norm. We also show that a multiplicative bound δ for the LP relaxation would imply $\delta \leq \frac{1}{2}$ (for $k = 2$), putting LP on par with other convex relaxations such as L2QP. Finally we characterize the equivalence classes of a (broader) class of reparameterizations, show their dimension, and how a basis can be used to generate potentially tigher relaxations.

Chapter 2

Markov random field

We review in this section the notion of Markov random field and its representation using the exponential family. Let $G = (V, E)$ be an undirected graph with nodes $i \in V$ and edges $(i, j) \in E$. We attach a discrete random variable $X_i \in \{0, ..., k-1\}$ to each node i with k the common number of labels (we can assume WLOG a uniform label set). We denote the concatenation of all variables as $X = (X_i)_{i \in V}$ and $X_A = (X_i)_{i \in A}$ the restriction to a subset $A \subset V$.

Definition 2.0.1 (Markov random field)**.** A Markov random field (MRF) models the joint probability distribution $p(X)$ of a collection of random variables having the Markov property w.r.t. a graph G: $\forall i$, $p(X_i | X_{V-\{i\}}) = p(X_i | x_{N_i})$, where N_i denotes the neighbors of i (Markov blanket).

The *Hammersley-Clifford theorem* states that for a positive MRF, $p(X)$ factorizes according to the cliques of the graph:

$$p(X) = \prod_{c \in C} \exp(\Phi_c(X)), \tag{2.0.1}$$

where C is the set of maximal cliques of the graph and Φ_c depends only on X_c. We represent X using a binary random variable $x = (x_{ia})$ with $x_{ia} = 1$ if $X_i = a$ and $\sum_a x_{ia} = 1$. We decompose the potential functions $\Phi_c(X)$ using a basis $\phi(x) = (\phi_\alpha(x))_{\alpha \in I}$ and a parameter vector θ, so that the MRF is represented with the *exponential family*:

$$p(x, \theta) \propto \exp(\langle \theta, \phi(x) \rangle) = \exp(\sum_{\alpha \in I} \theta_\alpha \phi_\alpha(x)) \tag{2.0.2}$$

In the remainder we assume that the positivity condition ($p(x) > 0$) is satisfied and that the MRF invovles interactions between at most pairs of variables (pairwise MRF)[1]. We will adress the higher order clique size in section 6. We can represent the MRF using the *canonical overcomplete representation*:

$$\phi(x) = \{x_{ia} : i \in V, a \in \{0, ..., k-1\}\} \cup \{x_{ia} x_{jb} : (i, j) \in E, (a, b) \in \{0, ..., k-1\}^2\} \tag{2.0.3}$$

[1] General MRFs can be converted to that form, see [16]

Thus, any pairwise positive MRF defined over G can be represented via some $\theta \in \mathbb{R}^d$ with $d = k|V| + k^2|E|$. Note, it is no longer a basis because of linear constraints such as $\sum_a x_{ia} = 1$. An important consequence is that an MRF can be represented equivalently by a whole subspace of exponential parameters.

2.1 Maximum Aposteriori (MAP) inference

The MAP inference problem is to maximize $p(x, \theta)$ over all feasible (discrete) assignments:

$$\phi_\infty(\theta) = \max_{x \text{ feasible}} \langle \theta, \phi(x) \rangle \tag{2.1.1}$$

Computing MAP estimates is the central problem in many applications, such as image processing, segmentation, labeling. In general finding the exact MAP is NP-hard, but there are important cases where the problem is tractable, such as for trees[5], for graphs with bounded tree-width, for graphs with a single cycle and binary variables[17], for supermodular potentials with binary variables[2], see section 8.1.

2.2 Other related inference and learning problems

There are other important inference problems besides MAP estimation, such as computation of marginals, max-marginals, and the partition function. In general those problems are equally intractable and require approximate solutions, but there are strong parallels between those problems. The max-product message passing algorithm for computation of max-marginals has its exact analog, sum-product, for computation of marginals. Notable tractable cases include trees, planar graphs with no interaction field and binary variables[18]. The latter paper provides another interesting parallel between estimation of max-marginals or MAP using convex combinations of trees on the one hand, and estimation of marginals or the partition function using convex combinations of planar graphs.

Several learning tasks are intimately coupled with inference, and certain tractable models such as Associative Markov Networks[19] allow for effective max-margin discriminative learning. For intractable models, the approximate MAP estimation we will review can provide the basis to develop efficient learning algorithms.

Chapter 3

Preliminaries

3.1 Exact formulations and other non-convex approximation algorithms

Before introducing the convex relaxations, we mention the existence of a general algorithm for exact MAP estimation, the junction tree algorithm. The algorithm works in essentially 3 steps: 1) triangulate the graph, 2) compute a junction tree, and 3) perform belief propagation on the tree, which computes the MAP. The applicability of junction tree algorithm depends on the tree-width, or size of the largest clique obtained in the junction tree. For certain non-tree graphs such as $k - fan$ graphs, the algorithm is the recommended solution. We are concerned here with the general case, for which junction tree is intractable.

There are a number of proposed strategies and heuristics for approximating the MAP, such as Iterative Conditional Modes[20], Relaxation Labeling, branch and bound, and other search algorithms. Those algorithms have either no approximation guarantee (local minima) or no bound on the running time. Instead we will focus on *convex approximations*, that optimize a convex upper bound of the problem. Those algorithms have both *approximation guarantees* and a *polynomial-time complexity*. We will see that, while off-the-shelf solvers can be used for the resulting convex program, in practice it is judicious to develop specialized algorithms such as message passing, fixed point updates or Sequential Minimal Optimization (SMO) to name a few techniques.

3.2 Convexification of the objective vs convexification of the domain

There are a number of proposed convex relaxations to the MAP problem: linear programming, semidefinite programming, second-order cone programming, convex quadratic programming, spectral relaxation, L2QP relaxation (equivalent to a geometric program). We distinguish two broad categories according to whether the cost function is quadratic (section 4) or linear (section 5). We will see that in fact relaxations in the first category all compute a convex upper-bound of the objective. On the other hand, relaxations in the second category all compute a hierarchy of successive convex upper bounds of the feasible set using lift-and-project reformulation: marginal polytope $MARG(G)$, semidefinite cone $SDP(G)$, second order cone $SOCP(G)$, local marginal polytope $LOCAL(G)$.

We present here some of the main recurring themes underlying the convex relaxations for the MAP problem.

3.3 Reparameterization

As noted previously, the exponential family for MRF is an overcomplete representation: there are multiple ways to encode the same cost function. We define the following equivalence relation, denoted as reparameterization:

$$\theta \equiv \theta' \iff \forall x \text{ feasible}, \langle \theta, \phi(x) \rangle = \langle \theta', \phi(x) \rangle \tag{3.3.1}$$

The flexibility offered by reparameterization is at the core of a number of the algorithms we will present: L2QP, CQP, spectral relaxation, dual formulation of the LP relaxation. In particular, we will see that tree-reweighted message-passing updates correspond to reparameterizations. A natural question that arises is whether or not a particular relaxation is invariant to reparameterizations; we will address this question is section 8.6.

Another important question concerns the equivalence classes of this relation. We have characterized them precisely in [21] and showed that they are subspaces of dimension $|V|(k^2 + k + 2)/2$, where we also allow terms of the form: $\theta_{ia,ib}$. Thus any reparameterization can be uniquely represented by a set of $|V|(k^2 + k + 2)/2$ free parameters, which correspond to Lagrangian dual variables in our formulation [21]. We showed how we can recover L2QP, CQP, spectral relaxation and other ones simply by optimizing an upper bound over different subsets of those free parameters.

3.4 Convexity of the MAP as function of the parameters

A central idea behind tree relaxation [5] and dual decomposition for MRF [22] is that the MAP value is a convex function of the parameters θ, allowing one to decompose the MRF into smaller, tractable subproblems for which we can compute the MAP, and then combine the solutions to approximate the original MAP. We will explore thos ideas in section 7, and show the connection with the above mentioned LP relaxation. Interestingly, the same idea is used in [18] for the problem of estimating the partition function or the marginals, but planar graphs are used as the tractable building blocks instead of trees.

3.5 Criteria to assess the different relaxations

We will analyse the different relaxations in terms of optimality guarantees, bounds on the quality of the approximation, relative dominance relations and importance of each particular constraint. We will also adress numerical issues such as time/space complexity and convergence properties.

Chapter 4

Quadratic relaxations to the MAP

4.1 Integer Quadratic Programming formulation (IQP)

We represent the constraint $\sum_a x_{ia} = 1$ as $Cx = 1$ for some matrix C, and introduce the $nk \times nk$ matrix W as $W_{iajb} = \theta_{ia,jb}$ and the $nk \times 1$ vector V as $V_{ia} = \theta_{ia}$. WLOG, we can assume W symmetric. The MAP problem is equivalent to the following Integer Quadratic Programming (IQP):

$$\boxed{\max \epsilon(x) = x^{\mathsf{T}} W x + V^{\mathsf{T}} x, \quad \text{s.t.} \quad Cx = 1, x \in \{0,1\}^{nk} \qquad (4.1.1)}$$

In general this IQP is NP-hard, and approximate solutions are needed. An interesting yet counterintuitive fact is that *we can remove the discrete constraint without changing the problem*, as we shall see in the next section. First, let us introduce some notations.

Definition 4.1.1 (Definitions)**.** We define the following constraint sets, where $n = |V|$:
$\Omega_a = \{x \in \mathbb{R}^{nk} : Cx = 1\}$ (**a**ffine set)
$\Omega_s = \Omega_a \cap \mathbb{R}_+^{nk}$ (standard **s**implex)
$\Omega_d = \Omega_a \cap \{0,1\}^{nk}$ (**d**iscrete set)
$\Omega_2 = \{x \in \mathbb{R}_+^{nk} : \sum_i x_{ia}^2 = 1\}$ (ℓ_2-sphere set)
Note, we have the following inclusions: $\Omega_d \subset \Omega_s \subset \Omega_a$ and $\Omega_d \subset \Omega_2$ defining relaxations to the feasible points of the IQP, Ω_d.

4.2 QP formulation

The QP relaxation relaxes the set Ω_d to Ω_s in (4.1.1):

$$\boxed{\max \quad \epsilon(x), \quad \text{s.t.} \quad Cx = 1, 0 \le x \le 1 \qquad (4.2.1)}$$

Proposition 4.2.1 (QP is equivalent to IQP)**.** *Suppose $W_{iaib} = 0, \forall i, a, b$, all other entries in W being unconstrained. Then from any $x \in \Omega_s$, we can construct efficiently $x_d \in \Omega_d$ such that $\epsilon(x_d) \ge \epsilon(x)$. As a corollary, $\max_{x \in \Omega_s} \epsilon(x) = \max_{x_d \in \Omega_d} \epsilon(x_d)$ and (4.2.1) is equivalent to (4.1.1).*

See [8, 9] for a proof or [10] for a more general result. This proposition will be used to prove optimality bounds.

4.3 Convex quadratic programming relaxation (CQP)

In [9], the authors approximate the QP (4.2.1) with a Convex quadratic programming relaxation (**CQP**) by using the following reparameterization: they replace (W, V) with $(W - diag(D), V + D)$ where D is a vector such that $W - diag(D) \preceq 0$. They propose $D = |W|1$ (row-sum of the absolute values) to make $diag(D) - W$ diagonally dominant (ensuring $W - diag(D) \preceq 0$), but other choices are possible. The associated cost function is:

$$\epsilon_{CQP}(x) = x^{\mathsf{T}}(W - diag(D))x + V + D^{\mathsf{T}}x = \epsilon(x) + \sum_{\alpha} D_\alpha(x_\alpha - x_\alpha^2) = \epsilon(x) + \sum_{\alpha,\beta} |W_{\alpha,\beta}|(x_\alpha - x_\alpha^2) \tag{4.3.1}$$

the last equality follows from the particular choice of D. It is indeed an upper bound on $\epsilon(x)$: for $x \in \Omega_s$, $x_\alpha - x_\alpha^2 \geq 0$ and $D_\alpha \geq 0$. The resulting quadratic program can be solved with Iterative Conditional Modes[20], or more efficiently with projected conjugate gradient ascent[23]. Note, the conditions of proposition 4.2.1 do not apply anymore since we have subtracted terms from the diagonal.

In [9] the authors claim that proposition 4.2.1 can be used to show the MAP problem is solvable in polynomial time if the edge parameter matrix W is negative definite. We show that this is in fact *impossible*: under the conditions of 4.2.1, by looking at all 2×2 principal minors, we see that in fact the only semidefinite negative matrix with 0 on the diagonal is 0.

A discrete solution can nevertheless be recovered efficiently by applying proposition 4.2.1 to the original parameterization (W, V) starting from the continuous solution of $(W - diag(D), V + D)$. This gives the following additive bound:

Proposition 4.3.1 (additive bound for CQP). *Let x be the optimal solution for the convex QP parameterized with $(W - diag(D), V + D)$. Then one can efficiently compute a discrete solution x_d such that $\epsilon(x_d) \geq \max_{\Omega_d} \epsilon - \frac{1}{4}D1$. When $D = |W|1$, the additive bound is $\frac{1}{4}||W||_1$ (tight bound).*

See [9] for the proof.

4.3.1 Improving the additive bound

Note, there are other possible choices for D than the one suggested in [9], for example $D = \lambda_{\max}[W]1$ leads to and additive bound of $\frac{n}{4}\lambda_{\max}[W]$. In our experiments, this choice provides a better bound when W is sparse and contains both positive and negative terms, which is a common case. Based on proposition 4.3.1, the best possible bound can be formulated as:

$$\min_D D^{\mathsf{T}}1 \quad \text{s.t.} \quad W - diag(D) \preceq 0 \tag{4.3.2}$$

whose dual is the following SDP, where strong duality holds:

$$\max_X tr(WX) \quad \text{s.t.} \quad X \succeq 0, diag(X) = I \tag{4.3.3}$$

4.4 L2QP relaxation

In [7, 8], the authors relax the IQP as:

$$\max \epsilon(x) \quad \text{s.t.} \quad \sum_a x_{ia}^2 = 1 \tag{4.4.1}$$

and make the assumption that W, V are nonnegative (we can always add a constant term to each $W_{ia,jb}, V_{ia}$ with $i \neq j$ without changing the discrete optimal configurations). Geometrically, the constraint set is changed from a product of simplices Ω_s to a product of ℓ_2 spheres Ω_2 (hence the name L2QP). We can instead reinterpret L2QP as a convex upper bound on $\epsilon(x)$: define $\epsilon_{L2QP}(x) = \epsilon(\sqrt{(x)})$ with the original constraint set $x \in \Omega_s$ (equivalent to $\sqrt{(x)} \in \Omega_2$). Then $\epsilon_{L2QP}(x) \geq \epsilon(x)$ since W, V are nonnegative and $\sqrt{(x)} \geq x$ on Ω_s. Furthurmore, computing the Hessian of each monomial reveals that ϵ_{L2QP} is concave. Note, this is also equivalent to a geometric program as we can safely relax the equality constraint to an inequality, and because ϵ_{L2QP} is a posynomial.

4.4.1 Optimization using Lagrangian duality

Introducing lagrange multipliers λ_i for each site constraint $\sum_a x_{ia}^2 = 1$, the lagrangian is:

$$L(x, \lambda) = \epsilon(x) + \sum_i \lambda_i (\sum_a x_{ia}^2 - 1) \tag{4.4.2}$$

By linear independance constraint qualification we have the following KKT conditions: $\exists x^* \in \Omega_2, \exists \lambda \in \mathbb{R}^n$:

$$\forall ia, \frac{d\epsilon}{dx_{ia}}(x^*) + 2\lambda_i x_{ia}^* = 0 \tag{4.4.3}$$

Since $\sum_a x_{ia}^{*2} = 1$ we can eliminate λ_i and x^* must satisfy the following fixed point equation:

$$x^* = p_{\Omega_2}[Wx^* + \frac{1}{2}D] \tag{4.4.4}$$

where $p_{\Omega_2}[x]$ is the orthogonal projection of x onto Ω_2, defined as $p_{\Omega_2}[x]_{ia} = \frac{x_{ia}}{\sqrt{\sum_a x_{ia}^2}}$ (site-wise ℓ_2 normalization). Notice the strong similarity with the fixed-point equation used in the power method:

$$x^* = p_{\text{unit sphere}}[Wx^*] \tag{4.4.5}$$

This is not surprising, since the (normalized) leading eigenvector of a symmetric matrix W maximizes $x^\mathsf{T} W x$ subject to a (single) unit-sphere constraint. Moreover, we have an analog of Perron-Frobenius theorem which guarantees convergence of the fixed point iterates (4.4.5) to a *unique* point $x \geq 0$ in the unit sphere when W is nonnegative and irreducible:

Proposition 4.4.1. *When W is irreducible (and still assuming W, V nonnegative),* (4.4.4) *has a unique fixed-point, which is the unique maximizer of* (4.4.1)*.*

This is a special case of theorem 5 in [7].

4.4.2 Discretization and optimality bounds

We can compute a multiplicative bound using the following: let $x^* = \arg\max_{\Omega_2} \epsilon$, and $x_s = p_{\Omega_s}[x^*]$ where $p_{\Omega_s}[x]$ is a projection of x onto Ω_s, defined as $p_{\Omega_s}[x]_{ia} = \frac{x_{ia}}{\sum_a x_{ia}} \forall x > 0$ (site-wise ℓ_1 normalization). As shown in [8], x_s verifies $\epsilon(x_s) \geq \frac{1}{k}\epsilon(x^*) \geq \frac{1}{k}\max_{\Omega_d} \epsilon$. Finally using the rounding scheme of proposition 4.2.1, we can compute efficiently $x_d \in \Omega_d$ with $\epsilon(x_d) \geq \frac{1}{k}\max_{\Omega_d} \epsilon$.

4.5 Spectral Relaxation

In [10] we proposed the following relaxation:

$$\epsilon_{spectral}(x) = \frac{\epsilon(x)}{D(x)} \tag{4.5.1}$$

$$D(x) = \frac{x^\intercal x + \beta}{n + \beta} \tag{4.5.2}$$

with some constant $\beta > 0$. We showed that on Ω_s, $\epsilon_{spectral}(x) \geq \epsilon(x)$; furthermore we converted $\max_{x \in \Omega_a} \epsilon_{spectral}(x)$ to an equivalent eigenvalue problem, whose dual is convex and where strong duality holds. This also interprets $\epsilon_{spectral}(x)$ as a convex upper bound on $\epsilon(x)$ (as long as the maximizer is ≥ 0 to stay in Ω_s, see [10]). We also proposed data-dependant and data-independant optimality bounds and showed that they exactly matched the ones for L2QP.

Chapter 5

Linear relaxations to the MAP

We can rewrite the QP (4.2.1) equivalent to the MAP with a (linear) matrix inner product: $\epsilon(x) = x^{\mathsf{T}}Wx + V^{\mathsf{T}}x = X \bullet W + V^{\mathsf{T}}x = \epsilon(x, X)$ (by abuse of notation) with $X_{ia,jb} = x_{ia}x_{jb}$ $\forall(i,j) \in E$. Define $M_{indep}(G)$ as:

$$M_{indep}(G) = \{(x, X) : x \in \Omega_s, X_{ia,jb} = x_{ia}x_{jb}\forall(i,j) \in E\} \tag{5.0.1}$$

All the relaxations we present below relax the non-convex constraint $X_{ia,jb} = x_{ia}x_{jb}$. This results in a hierarchy of feasible sets with increasingly looser approximations. We also have $\phi(\Omega_d) \subset M_{indep}(G)$ as can be verified by taking $X_{ia,jb} = x_{ia}x_{jb} = \phi_{ia,jb}(x)$ for any $x \in \Omega_d$.

5.1 Linear Programming on the Marginal Polytope

Given some MRF $p(\cdot) = p(\cdot, \theta)$ over G, we define the marginal probabilities for each node and each edge as follows:

$$\begin{aligned} \mu_{ia} &= E_\theta[x_{ia}] = \sum_{x \in \mathcal{X}} p(x, \theta)x_{ia} && (5.1.1) \\ \mu_{ia,jb} &= E_\theta[x_{ia}x_{jb}] = \sum_{x \in \mathcal{X}} p(x, \theta)x_{ia}x_{jb} && (5.1.2) \end{aligned}$$

More compactly, $\mu = E_\theta[\phi(x)] \in \mathbb{R}^d$ with d the dimension of θ. We define the *marginal polytope* $MARG(G)$ as the set of all feasible marginals μ computed from some MRF $p(\cdot) = p(\cdot, \theta)$ over G:

$$MARG(G) = \{\mu \in \mathbb{R}^d : \exists\theta \in \mathbb{R}^d : \mu = E_\theta[\phi(x)]\} \tag{5.1.3}$$

By definition, $MARG(G)$ is also the convex hull of $\{\phi(x) : x \in \mathcal{X}\}$, hence the name (convex) polytope. We have the following results, which we generalize in section 6:

$$\begin{aligned} M_{indep}(G) &\subset MARG(G) && (5.1.4) \\ \max_{x \in \Omega_d} \epsilon(x) &= \max_{\mu \in MARG(G)} \langle \theta, \mu \rangle && (5.1.5) \end{aligned}$$

According to the Minkowski-Weyl theoremit can be written as an intersection of half-spaces: $MARG(G) = \{\mu \in \mathbb{R}^d : A\mu \leq b\}$ for some constraint matrix A and vector b. A, b are hard to express in general (when G is arbitrary), and so this formulation is mostly of theoretical importance.

5.2 Linear programming relaxation (LP)

In the linear programming formulation[14, 15] we only keep a polynomial sized (in $|V|, |E|$ and the number of labels) number of affine constraints (equality or inequality), denoted as $AFFINE(G)$. One commonly used set of constraints, used in [5], define *exhaustive* local consistency constraints on edges of the graph G:

$$LOCAL(G) = \{(x, X) : X \geq 0, \sum_a x_{ia} = 1, \sum_a X_{ia,jb} = x_{ia}, \sum_b X_{ia,jb} = x_{jb}\} \quad (5.2.1)$$

We can interpret (x, X) as *pseudomarginals*, since they approximate elements in $MARG(G)$, and the last 2 constraints as marginalization constraints. An important result coming from the junction tree decomposition is that for trees, $MARG(G) = LOCAL(G)$, but in general $MARG(G) \subsetneq LOCAL(G)$.

There are at least two ways to solve the resulting linear program: one is via a generic LP solver such as mosek, which scales to mid-sized instances. Another is via message-passing algorithms, which attempt to solve the dual of the LP, as we will see in section 7.

5.3 Semidefinite programming relaxation (SDP)

The **SDP** relaxation [11] attempts to tighten the LP relaxation. We introduce a symmetric matrix X to represent the terms $X_{ia,jb} = x_{ia}x_{jb}$ and relax the non-convex constraint $X = xx^\mathsf{T}$ with the semidefinite constraint $X \succeq xx^\mathsf{T}$. Using Schur's complement, this amounts to:

$$\begin{bmatrix} X & x \\ x^\mathsf{T} & 1 \end{bmatrix} \succeq 0 \quad (5.3.1)$$

Note, we have introduced variables of the form $X_{ia,jb}$ for $(i, j) \notin E$ in this representation, and so we introduce the projection $p_G(X) = (X_{ia,jb} : (i, j) \in E)$ for comparison with other relaxations. We define $SDP(G)$ as the set of $(x, p_G(X))$ where (x, X) satisfies (5.3.1) (with X symmetric). We have:

$$MARG(G) \subset SDP(G) \cap AFFINE(G) \subset AFFINE(G) \quad (5.3.2)$$

The semidefinite program $\max_{(x,X) \in SDP(G) \cap AFFINE(G)} X \bullet W + V^\mathsf{T} x$ offers a good approximation of the MAP solvable in polynomial time, but requires expensive SDP solvers that often don't scale beyond a thousand variables. Another main drawback is that the number of variables is squared (compared to the QP formulation), and cannot take advantage of the sparsity of the graph, as opposed to the LP relaxation.

5.4 Second-order cone programming relaxation (SOCP)

The **SOCP** relaxation [12, 13] proposes a more efficient method than SDP relaxation by further relaxing $X \succeq xx^\mathsf{T}$ to $X \bullet S \geq (xx^\mathsf{T}) \bullet S = x^\mathsf{T} S x$ for a suitable polynomial-sized set of positive semidefinite symmetric matrices $S \in \mathbb{S}_+$. Using the fact $S = UU^\mathsf{T}$ for some U we can rewrite the constraint as:

$$\begin{aligned} x^\mathsf{T} U U^\mathsf{T} x &\leq X \bullet S && (5.4.1)\\ \iff ||U^\mathsf{T} x||^2 &\leq X \bullet S && (5.4.2)\\ \iff || \begin{bmatrix} 1 - X \bullet S \\ 2U^\mathsf{T} x \end{bmatrix} || &\leq 1 + X \bullet S && (5.4.3) \end{aligned}$$

where the last inequality shows the equivalence to an explicit SOCP constraint. In [12, 13] the authors choose the set of SOCP constraints given by matrices U as follows:

$$(U) = \{e_\alpha, e_\alpha + e_\beta, e_\alpha - e_\beta : \alpha = ia, \beta = jb, (i,j) \in E\} \tag{5.4.4}$$

where (e_α) is the canonical basis. The corresponding SOCP constraints are:

$$\begin{aligned} x_\alpha^2 &\leq X_{\alpha,\alpha} && (5.4.5)\\ (x_\alpha + x_\beta)^2 &\leq X_{\alpha,\alpha} + X_{\beta,\beta} + 2X_{\alpha,\beta} && (5.4.6)\\ (x_\alpha - x_\beta)^2 &\leq X_{\alpha,\alpha} + X_{\beta,\beta} - 2X_{\alpha,\beta} && (5.4.7) \end{aligned}$$

We can simplify these by using the constraint $X_{\alpha,\alpha} = x_\alpha$ (representing $x_\alpha^2 = x_\alpha$ for the discrete case) to eliminate the diagonal terms $X_{\alpha,\alpha}$, and then by eliminating the first constraint, since it is equivalent to $x \in [0,1]$. We finally obtain, after algebraic manipulations:

$$|X_{\alpha,\beta} - x_\alpha x_\beta| \leq \frac{1}{2}(x_\alpha - x_\alpha^2 + x_\beta - x_\beta^2) \tag{5.4.8}$$

We define $SOCP(G)$ be the set of (x, X) satisfying those constraints. With those local constraints we have reduced the number of variables from $(|V|k)^2$ down to $|V|k + |E|k^2$, using the same variables as in the LP relaxation[1].

5.5 Additional affine constraints

In any of the linear relaxations (LP, SDP, SOCP) we can also add the following triangular inequalities to $AFFINE(G)$ to tighten the relaxation: $\forall \alpha, \beta, \gamma$,

$$\begin{aligned} x_\alpha + x_\beta + x_\gamma &\leq X_{\alpha,\beta} + X_{\beta,\gamma} + X_{\gamma,\alpha} \\ x_\beta &\geq X_{\alpha,\beta} + X_{\beta,\gamma} - X_{\gamma,\alpha} \quad \text{(and circular permutations)} \end{aligned} \tag{5.5.1}$$

[1] Note, this is a simpler, but equivalent presentation compared to [13], in which the authors used the range [-1,1] instead of [0,1].

They can be verified using a truth table for binary variables, and are equivalent to the ones used in [13] with the range [-1,1]. We can define the following sets of affine constraints for $AFFINE(G)$:

$$\begin{aligned}
LOCAL_1(G) &= \{(x,X) : x \in \Omega_s\} && (5.5.2)\\
LOCAL_2(G) &= LOCAL(G) \quad \text{see (5.2.1)} && (5.5.3)\\
LOCAL_3(G) &= LOCAL_2(G) \cap \{(x,X) : \text{(5.5.1) is satisfied }\} && (5.5.4)\\
LOCAL_{+\infty}(G) &= MARG(G) \quad \text{see (5.1.3)} && (5.5.5)
\end{aligned}$$

In each case $LOCAL_p(G)$ only considers equalities/inequalities supported by at most p vertices in the graph. We have the following inclusions of polytopes $LOCAL_{+\infty}(G) \subset LOCAL_1(G) \subset LOCAL_3(G) \subset LOCAL_2(G) \subset LOCAL_1(G)$. We will use those definitions in section 8 when comparing the different relaxations.

Chapter 6

Generalization to higher order clique potentials

We generalize some results for the most general case of MRF with higher order clique potentials. We can write the MAP problem as the following integer program with polynomial cost function:

$$\max_{x \in \Omega_d} \epsilon(x) \tag{6.0.1}$$

$$\epsilon(x) = \langle \theta, \phi(x) \rangle = \sum_{\alpha \in I} \theta_\alpha \phi_\alpha(x) \tag{6.0.2}$$

$$\tag{6.0.3}$$

where $\phi_\alpha(x)$ is a monomial of degree $deg(\phi_\alpha(x))$ that involves at most one term x_{ia} per vertex i: we write $\alpha = \alpha_1 ... \alpha_m$ if $\phi_\alpha(x) = x_{\alpha_1} ... x_{\alpha_m}$ and $m = deg(\phi_\alpha(x))$. The marginal polytope extends its definition in a straightforward manner:

$$MARG(G) = \{\mu \in \mathbb{R}^d : \exists \theta \in \mathbb{R}^d : \mu = E_\theta[\phi(x)]\} \tag{6.0.4}$$

with $d = |I|$. We also define $M_{indep}(G)$ as the subset of $MARG(G)$ corresponding to independent variables:

$$M_{indep}(G) = \{\mu \in \mathbb{R}^d_+ : \sum_a \mu_{ia} = 1, \mu_{\alpha_1 ... \alpha_m} = \mu_{\alpha_1} ... \mu_{\alpha_m}\} \tag{6.0.5}$$

We verify indeed that $M_{indep}(G) \subset MARG(G)$ as follows: take $\theta_{ia} = \mu_{ia}$ and $\theta_{ia,jb} = 0$, so that the random variables are independant. Then it is easy to see that $E_\theta[x_{\alpha_1} ... x_{\alpha_m}] = \mu_{\alpha_1} ... \mu_{\alpha_m}$. By definition, we also have $M_{indep}(G) = \phi(\Omega_s)$.

6.1 Polynomial formulation

We relax (6.0.1) as follows and prove a generalization of proposition 4.2.1.

$$\max_{x \in \Omega_s} \epsilon(x) \tag{6.1.1}$$

Proposition 6.1.1 ((6.0.1) and (6.1.1) are equivalent)**.** $\max_{x\in\Omega_d}\epsilon(x) = \max_{x\in\Omega_s}\epsilon(x)$ *and given a point* $x \in \Omega_s$ *we can efficiently construct* $x_d \in \Omega_d$ *with* $\epsilon(x) \leq \epsilon(x_d)$*.*

Proof.

$$\max_{x\in\Omega_d}\epsilon(x) = \max_{x\in\Omega_d}\langle\theta, \phi(x)\rangle = \max_{\mu\in MARG(G)}\langle\theta, \mu\rangle \tag{6.1.2}$$

since $MARG(G)$ can be seen as the convex hull of $\phi(\Omega_d)$ and a linear program is maximized at a vertex. Also,

$$\max_{x\in\Omega_s}\epsilon(x) = \max_{\mu\in M_{indep}(G)}\langle\theta, \mu\rangle = \max_{\mu\in MARG(G)}\langle\theta, \mu\rangle \tag{6.1.3}$$

since $\phi(\Omega_d) \subset M_{indep}(G) \subset MARG(G)$. Finally, $\max_{x\in\Omega_d}\epsilon(x) = \max_{x\in\Omega_s}\epsilon(x)$. Now suppose we are given $x \in \Omega_s$. We follow a rounding scheme similar to Iterative Conditional Modes: let $i \in V$ and partition x as $x = (x_i, y)$ with $x_i = (x_{ia})$. $x_i \to \epsilon(x_i, y)$ is a linear function of x_i, and is maximized at some extremal point. We assign that value to x_i and proceed with the other vertices until we obtain some $x_d \in \Omega_d$. By construction, $\epsilon(x_d) \geq \epsilon(x)$ □ □

6.2 Generalization of optimality bounds

We show the following proposition generalizing the results of section 4.4.2:

Proposition 6.2.1 ($\frac{1}{k^{p-1}}$ bound for nonnegative MRFs with size p cliques and k labels)**.** *When the coefficients θ are nonnegative (*i.e.*$\epsilon(x)$ is a posynomial), we can efficiently compute a feasible point $x_d \in \Omega_d$ such that $\epsilon(x_d) \geq \frac{1}{k^{p-1}}\max_{\Omega_d}\epsilon$, with k the maximum number of labels per node and p the maximal clique size.*

Proof. We assume here that θ is nonnegative (*i.e.*$\epsilon(x)$ is a posynomial). We relax the linear constraints as $\sum x_{ia}^p = 1$ where $p = \max_\alpha deg(\phi_\alpha(x))$ so that x lies on a product of ℓ_p spheres (denoted Ω_p). The resulting program is still concave (up to a change of variables $x \to x^p$) and can be solved efficiently. As shown in [7], when $\epsilon(x)$ is irreducible (a generalization of irreducibility for a matrix), there is a unique solution which is also the unique critical point of $\epsilon(x)$ on Ω_p, but we do not need unicity for the bound.

Let $x^* = \arg\max_{\Omega_p}\epsilon$ be such a point (not necessarily unique) and $x_s = p_{\Omega_s}[x^*]$, using the projection defined earlier. We have $\sum_a x_{ia}^* \leq k^{1-1/p}$ (since $\sum_a x_{ia}^{*3} = 1$) and for each monomial $\phi_{\alpha_1...\alpha_m}(x)$ we have

$$\phi_{\alpha_1...\alpha_m}(x_s) = x_{s,\alpha_1}...x_{s,\alpha_1} \geq \frac{x_{\alpha_1}^*}{k^{1-1/p}}...\frac{x_{\alpha_m}^*}{k^{1-1/p}} = \frac{\phi_{\alpha_1...\alpha_m}(x^*)}{(k^{1-1/p})^m} \tag{6.2.1}$$

Since we assumed the coefficients are nonnegative and $m \leq p$, we obtain: $\epsilon(x_s) \geq \epsilon(x^*)\frac{1}{(k^{1-1/p})^p} = \frac{1}{k^{p-1}}$. Finally, using proposition 6.1.1 we can efficiently compute $x_d \in \Omega_d$ such that $\epsilon(x_d) \geq \epsilon(x_s) \geq \frac{1}{k^{p-1}}\max_{\Omega_d}\epsilon$ □ □

Chapter 7

Tree Relaxation, Message Passing and Lagrangian Duality

7.1 Upper bounds via convex combinations of trees

One key property of $\phi_\infty(\theta)$ is its convexity (as a maximum of linear functions). Let (θ^i) be a collection of exponential parameters and (ρ^i) be a collection of non-negative weights with $\sum_i \rho^i\theta^i = \theta$. By Jensen's inequality we have the following upper bound:

$$\phi_\infty(\theta) \leq \sum_i \rho^i \phi_\infty(\theta^i) \tag{7.1.1}$$

This bound is only useful when the parameters θ^i lead to tractable computations. We seek here the best (*i.e.*smallest) possible upper bound involving convex combinations of tree-structured exponential parameters, for which we can compute each MAP easily using dynamic programming.

We now introduce some notation to make this precise. Let T be a spanning tree defined over G with edges $E(T)$; we define $I(T)$ as the indices corresponding to the tree T and $\mathcal{E}(T)$ as the set of tree-structured MRF.

$$\begin{aligned} I(T) &= \{ia\} \cup \{(iai, jb) : (i,j) \in E(T)\} & (7.1.2)\\ \mathcal{E}(T) &= \{\theta(T) \in \mathbb{R}^d : \theta_\alpha = 0 \; \forall \alpha \in I \backslash I(T)\} & (7.1.3)\end{aligned}$$

Let $\mathcal{T}$ be a set of spanning trees such that $E = \cup_{T\in\mathcal{T}} E(T)$ and ρ be a positive probability distribution over $\mathcal{T}$. We define the edge appearance probability as $\rho_e = Pr_\rho[e \in T] > 0$ and the set of $\rho - reparameterizations$ as:

$$A_\rho(\theta) = \{(\theta(T))_{T\in\mathcal{T}} : \theta(T) \in \mathcal{E}(T),\; E_\rho[\theta(T)] = \sum_{T\in\mathcal{T}} \rho(T)\theta(T) = \theta\} \tag{7.1.4}$$

Since $\forall e \in E \rho_e > 0$ (by construction), we have $A_\rho(\theta) \neq \emptyset$.

7.2 Tightness of the upper-bound

It follows from (7.1.1) that for $(\theta(T)) \in A_\rho(\theta)$,

$$\phi_\infty(\theta) \leq \sum \rho(T)\phi_\infty(\theta(T)) \tag{7.2.1}$$

We are interested in the cases where the upper bound is tight. Define the collection of optimal configurations as:

$$OPT(\theta) = \arg\max_{x \in \Omega_d} \langle \theta, \phi(x) \rangle \tag{7.2.2}$$

The following proposition shows that when the so-called tree agreement condition holds, we can solve the original MAP problem by looking at configurations which are optimal for every tree $T \in \mathcal{T}$.

Proposition 7.2.1 (Tree agreement)**.** *For* $(\theta(T)) \in A_\rho(\theta)$*, we have:*

$$\cap_T OPT(\theta(T)) \subset OPT(\theta) \tag{7.2.3}$$

with equality and the bound (7.2.1) *is tight* iff *the LHS is non-empty (in which case there is also equality in* (7.2.3)*).*

Proof. (7.2.3) follows from (7.2.1). Let $x^* \in OPT(\theta)$. We can rewrite (7.2.1) as:

$$0 \leq \sum \rho(T)\phi_\infty(\theta(T)) - \phi_\infty(\theta) = \sum \rho(T)[\phi_\infty(\theta(T)) - \langle \theta(T), \phi(x^*) \rangle]$$

with equality only when $\langle \theta(T), \phi(x^*) \rangle = \phi_\infty(\theta(T)) \forall T \in \mathcal{T}$ □ □

We present two possible approches to minimizing the upper bound (7.2.1). The first one is based on direct minimization of (7.2.1) using Lagrangian duality leading to an LP. The second one is based on message-passing algorithms that seek to find a $\rho -$ *reparameterization* for which tree-agreement holds. The fixed-points of those algorithms correspond to optimal solutions of the LP when the bound is tight.

7.3 Lagrangian duality, tree relaxation and linear programming

We prove in this section the following fundamental results:

$$\begin{aligned} \phi_\infty(\theta) &\leq \min_{(\theta(T)) \in A_\rho(\theta)} E_\rho[\phi_\infty(\theta(T))] && (7.3.1) \\ \phi_\infty(\theta) &= \max_{\mu \in MARG(G)} \langle \theta, \mu \rangle && (7.3.2) \\ \min_{(\theta(T)) \in A_\rho(\theta)} E_\rho[\phi_\infty(\theta(T))] &= \max_{\mu \in LOCAL(G)} \langle \theta, \mu \rangle && (7.3.3) \end{aligned}$$

Here are a few important observations:

- (7.3.3) expresses the Lagrangian dual of the upper bound as an LP

- (7.3.3) shows that the particular choice of the tree collection $\mathcal{T}$ and distribution ρ does not affect the upper bound (as long as the edge appearance probabilities are all positive)
- even though (7.3.2) also expresses the MAP as an LP, the number of constraints for MARG(G) make it intractable in general
- for trees, we have equality in (7.3.1) because MARG(G)=LOCAL(G), but there are non-tree cases where equality holds as well

7.3.1 LP for the exact MAP

We give here a simpler proof of (7.3.2): $\max_{x\in\Omega_d}\langle\theta,x\rangle = \max_{\mu\in MARG(G)}\langle\theta,\mu\rangle$ because MARG(G) is the convex hull of Ω_d and an LP is maximized at a vertex.

7.3.2 LP for the upper bound

We prove the main result (7.3.3) expressing the Lagrangian dual of the upper bound as an LP. Let τ be the vector of Lagrange multipliers associated with the constraint $E_\rho[\theta(T)] = \theta$. The Lagrangian of the LHS of (7.3.3) is:

$$\begin{aligned} L_{\rho,\theta}((\theta(T)),\tau) &= E_\rho[\phi_\infty(\theta(T))] + \langle\tau, \theta - E_\rho[\theta(T)]\rangle && (7.3.4)\\ &= E_\rho[\phi_\infty(\theta(T)) - \langle\theta(T),\tau\rangle] + \langle\tau,\theta\rangle && (7.3.5)\end{aligned}$$

The dual function is:

$$\inf_{\theta(T)\in\mathcal{E}(T)} L_{\rho,\theta}((\theta(T)),\tau) = \langle\tau,\theta\rangle + E_\rho[\inf_{\theta(T)\in\mathcal{E}(T)} \phi_\infty(\theta(T)) - \langle\theta(T),\tau\rangle] \qquad (7.3.6)$$

Now, for a tree T we have

$$\phi_\infty(\theta(T)) = \max_{\tau\in LOCAL(G)}\langle\theta(T),\tau\rangle = \max_{\tau\in LOCAL(G,T)}\langle\theta(T),\tau\rangle \qquad (7.3.7)$$

where

$$LOCAL(G,T) = \{\mu\in\mathbb{R}^d_+ : \sum_a \mu_{ia} = 1, \sum_a \mu_{ia,jb} = \mu_{jb}, \sum_i \mu_{uv,ij} = \mu_{vj} \forall(u,v)\in E(T)\} \qquad (7.3.8)$$

Therefore the conjugate dual is the indicator function of LOCAL(G,T):

$$\sup_{\theta(T)\in\mathcal{E}(T)} \langle\theta(T),\tau\rangle - \phi_\infty(\theta(T)) = \begin{cases} 0 & \text{if } \tau\in LOCAL(G,T),\\ +\infty & \text{else}\end{cases} \qquad (7.3.9)$$

Finally, we have

$$\inf_{\theta(T)\in\mathcal{E}(T)} L_{\rho,\theta}((\theta(T)),\tau) = \begin{cases} \langle\tau,\theta\rangle & \text{if } \tau\in \cap_T LOCAL(G,T),\\ -\infty & \text{else}\end{cases} \qquad (7.3.10)$$

we can express the dual maximization problem using the fact $\cap_T LOCAL(G,T) = LOCAL(G)$:

$$\max_\tau \inf_{\theta(T)\in\mathcal{E}(T)} L_{\rho,\theta}((\theta(T)),\tau) = \max_{\tau\in LOCAL(G)}\langle\theta,\tau\rangle \qquad (7.3.11)$$

by strong duality, we obtain (7.3.3) □

7.4 Tree-reewighted message passing algorithms

Solving the LP for the upper bound is certainly feasible using off-the-shelf solvers, but can be expensive for large graphs. In this section we explore iterative message-passing algorithms that exploit the garphical structure of the problem. The fixed-points of those algorithms correspond to optimal dual solutions of the LP (when the relaxation is tight). For trees, these algorithms reduce to the ordinary max-product algorithm, and are otherwise different.

We define *max-marginals* ν for tree-structured distributions $\theta(T)$ as follows[1]:

$$\nu_{ia} = \max_{x:x_{ia}=1} \langle \theta(T), x \rangle \tag{7.4.1}$$

$$\nu_{ia,jb} = \max_{x:x_{ia}=1, x_{jb}=1} \langle \theta(T), x \rangle \tag{7.4.2}$$

Using the junction tree decompositionone can show that a tree-structured distributions can be factored in terms of max-marginals as follows:

$$\langle \theta(T), x \rangle = \sum \nu_{ia} x_{ia} + \sum (\nu_{ia,jb} - \nu_{ia} - \nu_{jb}) x_{ia} x_{jb} \tag{7.4.3}$$

this is also the factorization produced by max-product algorithm (in log-space). In turn, we define a function $\theta(T) = f_T(\nu)$ as follows: $\theta(T)_{ia} = \nu_{ia}$ and $\theta(T)_{ia,jb} = \nu_{ia,jb} - \nu_{ia} - \nu_{jb}$ if $(i, j) \in E(T)$ and $\theta(T)_{ia,jb} = 0$ otherwise. One can check whether a vector ν corresponds to valid max-marginals using local operations as follows:

Proposition 7.4.1. *A vector ν corresponds (up to an additive constant) to valid max-marginals for a tree T* iff $\forall (i, j) \in E(T), \nu_{ia} = \max_b \nu_{ia,jb} +$ *constant where the additive constant is independant of a.*

Given the max-marginal factorization one can compute the MAP solution using the following local optimality conditions:

Proposition 7.4.2 (Local optimality)**.** *Let ν be a vector of valid max-marginals for a distribution represented by $\theta(T)$. Then $x^* \in OPT(\theta(T))$* iff*:*

$$x^*_{ia} = 1 \implies a \in \arg\max_{a'} \nu_{ia'} \ \forall i \tag{7.4.4}$$

$$x^*_{ia} = 1, x^*_{jb} = 1 \implies (a, b) \in \arg\max_{a'b'} \nu_{ia',jb'} \ \forall (i, j) \in E \tag{7.4.5}$$

$$\tag{7.4.6}$$

For graphs with cycles, the computation of max-marginals is intractable in general. We present here iterative algorithms that update a vector of so-called pseudo-max-marginals ν that approximate the real max-marginals so that:

1. $\forall T, (\theta(T)) = (f_T(\nu))$ is a $\rho - reparameterization$ of the orginial distribution θ

2. tree consistency: $\forall T, \nu$ are valid max-marginals for $f_T(\nu)$ (cf proposition 7.4.1).

The algorithms presented satisfy condition 1. at every iteration, and condition 2. upon convergence to a fixed-point.

[1] Our definition differs from [5] in that we use max-marginals in log-space, for consistency of notation

7.4.1 Direct updating of pseudo-max-marginals

Input: exponential parameters θ

1. Initialize the pseudo-max-marginals:

$$\nu^0_{ia} = \theta_{ia} \tag{7.4.7}$$

$$\nu^0_{ia,jb} = \frac{1}{\rho_{ij}}\theta_{ia,jb} + \theta_{ia} + \theta_{jb} \tag{7.4.8}$$

2. for $t = 0 ... +\infty$ until convergence:

$$\delta_{i\to j,b} = \max_a \nu^t_{ia,jb} \tag{7.4.9}$$

$$\nu^{t+1}_{ia} = \nu^t_{ia} + \sum_{j\in N(i)} \rho_{ij}(\delta_{j\to i,a} - \nu^t_{ia}) \tag{7.4.10}$$

$$\nu^{t+1}_{ia,jb} = (\nu^t_{ia,jb} - \delta_{i\to j,b} - \delta_{j\to i,a}) + \nu^{t+1}_{ia} + \nu^{t+1}_{jb} \tag{7.4.11}$$

Algorithm 1: Edge-based reparameterization updates

We show here that condition 1 is satisfied after each iteration. After initialization step, let $\theta(T)^0 = f_T(\nu^0)$:

$$\sum_T \rho(T)\theta(T)^0_{ia} = \sum_T \rho(T)\theta_{ia} = \theta_{ia} \tag{7.4.12}$$

$$\sum_T \rho(T)\theta(T)^0_{ia,jb} = \sum_{T:(i,j)\in E(T)} \rho(T)\frac{1}{\rho_{ij}}\theta_{ia,jb} = \theta_{ia,jb} \tag{7.4.13}$$

For the induction step we have, with $\theta(T)^{t+1} = f_T(\nu^{t+1})$ just after an update:

$$\sum_T \rho(T)\langle\theta(T)^{t+1}, x\rangle = \sum_{ia}(\nu^t_{ia} + \sum_{j\in N(v)} \rho_{ij}(\delta_{j\to i,a} - \nu^t_{ia}))x_{ia} \tag{7.4.14}$$

$$+ \sum_{ia,jb:(i,j)\in E} \rho_{ij}(\nu^t_{ia,jb} - \delta_{i\to j,b} - \delta_{j\to i,a})x_{ia}x_{jb} \tag{7.4.15}$$

$$= \sum_{ia} \nu^t_{ia}x_{ia} + \sum_{ia,jb} \rho_{ij}(\nu^t_{ia,jb} - \nu^t_{ia} - \nu^t_{jb})x_{ia}x_{jb} \tag{7.4.16}$$

We used the fact that $\sum_{ia}\sum_{j\in N(i)} \rho_{ij}(\delta_{j\to i,a} - \nu^t_{ia})x_{ia} = \sum_{iajb:(i,j)\in E} \rho_{ij}(\delta_{i\to j,b} + \delta_{j\to i,a} - \nu^t_{ia} - \nu^t_{jb}))x_{ia}x_{jb}$ since $x_{ia} = \sum_j x_{ia}x_{jb}$ and using both directions for each edge. Note also that this is valid for any expression of $\delta_{i\to j,b}$ (interpretable as a message). Finally, using the induction hypothesis we obtain

$$\sum_T \rho(T)\langle\theta(T)^{t+1}, x\rangle = \sum_T \rho(T)\langle\theta(T)^t, x\rangle = \langle\theta, x\rangle \tag{7.4.17}$$

We now show that condition 2 is satisfied at a fixed point $\nu = \nu^t = \nu^{t+1}$:

$$\delta_{i\to j,b} - \nu_{jb} = -(\delta_{j\to i,a} - \nu_{ia}) \tag{7.4.18}$$

therefore $\nu_{ia} = \max_b \nu_{ia,jb} + \mathit{constant}$ $\square$

Chapter 8

Comparison of convex relaxations

8.1 Optimality cases

In general, optimality conditions depend both on the topology of the graph and the numerical values on the potentials. The $LP - LOCAL(G)$ relaxation is tight in following important cases:

1. G is a tree

2. $\epsilon(x)$ is supermodular and $k = 2$, $i.e. W_{i0,j0} + W_{i1,j1} \geq W_{i0,j1} + W_{i1,j0}$

The first case forms the basis of TRW and dual decomposition for MRF. The second case forms the basis for graphcuts[2] and its multilabel extensions to α expansion moves and $\alpha - \beta$ swaps [24]. We also note there are exact algorithms for graphs with bounded tree-width (junction tree decomposition) and graphs containing a single loop and binary variables[17]. There are no such guarantees for $L2QP$, CQP, $spectralrelaxation$.

8.2 Relative dominance relationships

In order to compare relaxations we introduce the following notions. We say that relaxation A *domintes* relaxation B if for all problem instances, $\epsilon^*_A \leq \epsilon^*_B$ where ϵ^*_A denotes the optimum of relaxation A (implicitly depending on the problem instance). A *strictly domintes* B if A domintes B but B does not dominate A. A and B are equivalent if A domintes B B domintes A.

We have the following hierarchy of feasible sets:

$$\phi(\Omega_d) \subset M_{indep}(G) \subset MARG(G) \subset SDP(G) \cap AFFINE(G) \subset SOCP(G) \cap AFFINE(G) \tag{8.2.1}$$

where $AFFINE(G)$ is, again, any subset of affine constraints from the marginal polytope $MARG(G)$, for example $LOCAL(G)$, but we have seen that we can add other ones, such as triangular inequalities. In general all the inclusions are strict, but when G is a tree we have $MARG(G) = LOCAL(G)$. It follows that:

$$\epsilon^*_{IQP} = \epsilon^*_{QP} = \epsilon^*_{MARG(G)} \leq \epsilon^*_{SDP(G) \cap AFFINE(G)} \leq \epsilon^*_{SOCP(G) \cap AFFINE(G)} \leq \epsilon^*_{AFFINE(G)} \tag{8.2.2}$$

using optimizers on the respective sets; notably the last one being the Linear Programming relaxation on $AFFINE(G)$ ($LP - AFFINE(G)$). The two equalities come from proposition 4.2.1 and section 5.1. In general all the inequalities are strict, but obviously not when the LP relaxation is tight, see section 8.1.

A very interesting recent result from [25] shows that CQP is equivalent to $SOCP - LOCAL_1(G)$, which is the SOCP relaxation with $AFFINE(G) = LOCAL_1(G)$ as the affine constraint set (from section 5.5).

Proposition 8.2.1 (CQP is equivalent to SOCP with $LOCAL_1(G)$)**.** *$\forall x \in \Omega_s$, $\epsilon_{CQP}(x) = \max_{X:(x,X)\in SOCP(G)\cap LOCAL_1(G)} \epsilon(x,X)$. In particular, $\epsilon^*_{CQP} = \epsilon^*_{SOCP(G)\cap LOCAL_1(G)}$.*

We provide a more concise proof here: given $x \in \Omega_s$, the cost function $\epsilon(x,X)$ is linear in $X_{\alpha,\beta}$ (with coefficient $W_{\alpha,\beta}$) and there is a single box constraint for $X_{\alpha,\beta}$ given by (5.4.8). At the optimum we therefore have:

$$X_{\alpha,\beta} = x_\alpha x_\beta + sign[W_{\alpha,\beta}]\frac{1}{2}(x_\alpha - x_\alpha^2 + x_\beta - x_\beta^2) \qquad (8.2.3)$$

Linearly combining those equations we obtain:

$$\sum_{\alpha,\beta} W_{\alpha,\beta}X_{\alpha,\beta} = \sum_{\alpha,\beta} W_{\alpha,\beta}x_\alpha x_\beta + \sum_{\alpha,\beta} |W_{\alpha,\beta}|(x_\alpha - x_\alpha^2) \qquad (8.2.4)$$

and adding the linear term $\sum_\alpha V_\alpha x_\alpha$ to both sides we finally obtain:

$$\epsilon(x,X) = \epsilon_{CQP}(x) \quad \text{(using (4.3.1))} \qquad (8.2.5)$$

The conclusion follows taking $x^* = \arg\max_{\Omega_s} \epsilon_{CQP}$ □.

A second interesting result from [25] shows that $SOCP - LOCAL_1(G)$ is strictly dominated by $LP - LOCAL_2(G)$, and that the SOCP constraint (5.4.8) is implied by $LOCAL_2(G)$.

Proposition 8.2.2 ($LP - LOCAL_2(G)$ strictly dominates $SOCP - LOCAL_1(G)$)**.** *$LP - LOCAL_2(G)$ strictly dominates $SOCP - LOCAL_1(G)$, and in fact $LOCAL_2(G) \subset SOCP(G)$.*

In turn, this shows that $LP - LOCAL_2(G)$ strictly dominates CQP as well. For the proof, we first show, using the marginalization constraint that for $(x,X) \in LOCAL_2(G)$,

$$|x_\alpha - x_\beta| \leq x_\alpha + x_\beta - 2X_{\alpha,\beta} \qquad (8.2.6)$$

Next, the marginalization and positivity constraint also implies that:

$$x_\alpha + x_\beta \leq X_{\alpha,\beta} + 1 \leq 2X_{\alpha,\beta} + 1 \qquad (8.2.7)$$

Combining (8.2.6) and (8.2.8) we obtain:

$$2X_{\alpha,\beta} - 2x_\alpha x_\beta \leq (x_\alpha - x_\alpha^2 + x_\beta - x_\beta^2) \qquad (8.2.8)$$

We prove the lower bound similarly, and finally obtain (5.4.8). Furthermore, one can construct an MRF for which the optimizers of $SOCP - LOCAL_1(G)$ are outside of $LOCAL_2(G)$ □

8.3 Additive bounds on the quality of the approximation

In [25] the authors prove that we have the same additive bound for $LP - LOCAL_2(G)$ as for CQP, and propose an example to suggest that the bound is tight. However their example seems bogus (see their figure 1): since the graph has two nodes, it is a tree, so there is an integral solution to the LP (which is also the discrete optimal). The rounding scheme would not change that solution.

We suggest a slight modification of CQP to tighten their additive bound of $\frac{1}{4}||W||_1 = \frac{1}{4}\sum_{ij}|W_{i0,j0}| + |W_{i0,j1}| + |W_{i1,j0}| + |W_{i1,j1}|$ in the case $k = 2$. The new bound we obtain is:

$$\frac{1}{4}\sum_{ij}|W_{i0,j0} + W_{i1,j1} - W_{i0,j1} - W_{i1,j0}| \tag{8.3.1}$$

which we can interpret as the ℓ_1 deviation from modularity (recall that W represents a modular energy function if $\forall(i,j), W_{i0,j0} + W_{i1,j1} = W_{i0,j1} + W_{i1,j0}$). This bound is obviously tighter in general and has the significant advantage of being independant of reparameterizations, as one can show. Since the LP is invariant to reparameterizations, this new bound also applies to the LP relaxation. To prove this, we first reorder the indexes so as to decompose x, W, V as follows:

$$W = \begin{bmatrix} W_{00} & W_{01} \\ W_{10} & W_{11} \end{bmatrix}, \quad V = \begin{bmatrix} V_0 \\ V_1 \end{bmatrix}, \quad x = \begin{bmatrix} x_0 \\ 1 - x_0 \end{bmatrix} \tag{8.3.2}$$

We then rewrite the cost function using simple algebraic manipulations as:

$$x^\mathsf{T} W x + V^\mathsf{T} x = {x_0}^\mathsf{T}(W_{00} + W_{11} - W_{01} - W_{10})x_0 + {x_0}^\mathsf{T}(W_{01}1 + {W_{01}}^\mathsf{T}1 - 2W_{11}1) + 1^\mathsf{T} W_{11} 1 \tag{8.3.3}$$

Note, this is a simple matrix view of the transformation given in [2]. We can in fact interpret it as a reparameterization. Now we can apply the CQP algorithm to the new resulting quadratic program, and the bound (8.3.1) follows. We can see that this bound is invariant to reparameterizations, as the bound also applies to the reduced upper-left quadrant matrix following from this construction.

8.4 Multiplicative bounds

We assume in this section that W, V are nonnegative, otherwise there is no hope for a multiplicative bound (for example when the MAP is the only discrete solution with a positive outcome). We have seen that $L2QP$ provides a $\frac{1}{k}$ multiplicative bound, which is interesting since it does not depend on the parameters W, V, and we have generalized it to arbitrary clique size, giving a $\frac{1}{k^{p-1}}$ bound for cliques of size p. We have also shown in [10] that spectral relaxation provides a $\frac{1}{k}$ approximation guarantee whenever the maximizer of the relaxation is nonnegative (which is typically the case in our experiments).

Data-dependant multiplicative bounds have also been proposed for L2QP and spectral relaxation, as a function of the peakiness of the MAP. We proved in [10] that the bounds are in fact the same for L2QP and spectral relaxation (modulo the same caveat regarding nonnegativity of the maximizer for spectral relaxation). In particular, the bound converges to 1 as the distribution becomes more uniform.

There seems to be no multiplicative bound reported for the LP relaxation. We have numerically found examples with $k = 2$ for which the bound was approaching $\frac{1}{2}$ and we can probably prove that $\forall \eta > 0$, we can find an example for which the rounding scheme is less than $\frac{1}{2} - \eta$ times the discrete optimum, implying that if there is such a multiplicative bound, it is $\frac{1}{k}$ for $k = 2$ (those examples converge to a matrix for which the LP has an integral solution, but there are arbitrarily close examples for which the LP is fractional). We also found (rare) numerical examples with $k \geq 3$ for which the bound was > 2. We conjecture that the same multiplicative bound of $\frac{1}{k}$ holds for LP.

One can prove certain multiplicative bounds for the LP for special cases, such as Potts model or truncated linear pairwise potentials.

8.5 Ability to handle arbitrary clique size

Any of the lift-and-project relaxations (LP, SDP, SOCP) can naturally encode arbitrary clique sizes, with an increase in number of variables and computation time. In general, each clique of size p with k labels per node will require k^p marginals, one per joint assignment in that clique. The extension we proposed for the L2QP relaxation in section provides a much more compact representation, still using only $n \cdot k$ variables. The cost function is a polynomial (in fact posynomial) of degree p, and the relaxation can be shown equivalent to a geometric program (we can relax the ℓ_p sphere equality constraints to inequality constraints without changing the problem) and we have shown a multiplicative bound of $\frac{1}{k^{p-1}}$ in that case.

8.6 Invariance to reparameterization

One important aspect for a relaxation is the ability to produce the same answer regardless of the particular reparameterization of the cost function. We can see that CQP is not invariant, since the initial choice of D obviously depends on reparameterizations. We suggested a simple modification in section 8.3 to make CQP invariant in the case $k = 2$. L2QP is also not invariant to reparameterizations; in particular we need a reparameterization which makes the quadratic and linear part nonnegative (so as to obtain a posynomial).

One can show that the LP and SDP relaxations are invariant (indeed the dual variables correspond to reparameterizations).

8.7 Space/time complexity and convergence guarantees.

8.7.1 Space complexity

The relaxations we presented in section 4 do not change the number of variables of the original IQP formulation: they have $|V|k$ variables and same order of number of constraints. The LP and SOCP relaxations can both take advantage of the sparsity of the graph, and have $|V|k + |E|k^2$ variables and same order of number of constraints (more generally, with higher order terms we would have mk^p variables for a graph with m cliques of order p). Note that those relaxations can be tightened at the expense of increasing the number of constraints, for example by considering triangular inequalities as explained in section 5.5. The SDP formulation squares the number of variables ($(|V|k)^2$) as the SDP constraint is global and cannot take advantage of sparsity of the graph structure. In all those cases (except for the SDP formulation), the memory complexity is dominated by the problem instance (one needs to store the node and edge potentials). When the problem is structured however (potts model, truncated linear pairwise potentials, etc.), as is often the case for large-scale problems, the variables corresponding to marginals become the memory bottleneck, and the k^2 factor can become an issue.

8.7.2 Time complexity

Time complexity is an important consideration when deciding which algorithm to use, and there is a large body of research on designing specialized algorithms to improve complexity for particular cases, a discussion of which is beyond the scope of this report. Notable examples include dynamic graph cuts[26], improvements over $\alpha-$ expansion moves [27], linear-time generalized max-transform [28] to eliminate the k^2 factor in belief propagation updates, etc. Message passing algorithms [5, 29, 22] have also been widely used in practice to speed-up the LP relaxation. They can scale to large-scale problems and also avoid the k^2 memory factor observed in marginals.

There are also effective solvers for L2QP (using the fixed-point updates), CQP (SMO or conjugate gradient descent), and spectral relaxation (Lanczos method for eigenvalue computation) which allow to tackle large-scale problems. On the other hand, LP and SOCP scale to mid-size problems while SDP scales to smaller sized problems with a thousand variables.

8.7.3 Convergence guarantees

One of the main drawbacks of message passing algorithms such as TRW is the lack of general guarantees regarding convergence. TRW does not necessarily decrease the upper-bound at each iteration, and even worse, it may not converge at all. A sequential TRW variant (TRW-S) has been proposed in [29], with better convergence properties. In particular there is monotonic convergence to a local minimum satisfying the so-called weak tree agreement condition, less restrictive than the original tree agreement condition of [5]. Very recently, a dual decomposition technique has been applied to the MAP problem, where the problem is decomposed over a set of spanning trees. The resulting algorithm is a subgradient descent in the dual formulation, and the updates can be interpreted as a new message passing scheme. They have the attractive property of corresponding to a subgradient descent in the dual, and therefore come with much stronger convergence guarantees.

Chapter 9

Synthesis and Conclusion

9.1 Summary

We have shown how the MRF-MAP estimation problem, equivalent to an IQP, can be relaxed to an *equivalent* real-valued QP. The caveat is that the resulting QP is always non-convex except for the trivial case of an interaction-free MRF. We have presented a number of convex relaxations for the QP and showed they can be reformulated so as to fall in one of those two categories:

- (1) QP relaxations, involving a convex upper bound on the *objective* $\epsilon(x)$

- (2) lift-and-project relaxations, involving a linearization of the objective and a convex upper bound on the *domain* $X = xx^{\mathsf{T}}, x \in \Omega_s$

The lift-and-project operation transforms the non-convex objective into a linear (convex) one, while mapping the convex domain Ω_s to a non-convex one $\phi(\Omega_s) = M_{indep}(G)$. The different relaxations are summarized in table 9.1. The hierarchy of feasible sets is reproduced below:

$$\phi(\Omega_d) \subset M_{indep}(G) \subset MARG(G) \subset SDP(G) \cap AFFINE(G) \subset SOCP(G) \cap AFFINE(G) \tag{9.1.1}$$

where $AFFINE(G)$ is a set of local affine constraints, defined for example on nodes ($LOCAL_1(G)$), edges ($LOCAL_2(G)$) or triangles ($LOCAL_3(G)$), at the extreme on the entire graph ($LOCAL_\infty(G) = MARG(G)$). This yields the following dominance relations:

$$\epsilon^*_{IQP} = \epsilon^*_{QP} = \epsilon^*_{MARG(G)} \leq \epsilon^*_{SDP(G) \cap AFFINE(G)} \leq \epsilon^*_{SOCP(G) \cap AFFINE(G)} \leq \epsilon^*_{AFFINE(G)} \tag{9.1.2}$$

There are a few notable, if not surprising results coming from our analysis:

- (1) $\epsilon^*_{IQP} = \epsilon^*_{QP}$, *i.e.*the discrete constraint is redundant

- (2) $\epsilon^*_{QP} = \epsilon^*_{MARG(G)}$, *i.e.*the rank-1 constraint $X = xx^{\mathsf{T}}$ can be approximated by a finite (but very large) set of affine constraints

QP relax.	convex upper bound on the COST $\epsilon(x)$
L2QP	$\epsilon(x) \leq \epsilon(\sqrt(x))$
spectral	$\epsilon(x) \leq \epsilon(x)/(x^\top x/n)$
CQP	$\epsilon(x) \leq \epsilon(x) + \sum_\alpha (x_\alpha - x_\alpha^2)$
lift&project	linearize and convex upper bound on DOMAIN $X = xx^\top, x \in \Omega_s$
SDP	$X = xx^\top$ approximated as $X \succeq xx^\top$ + affine constraints
SOCP	$X \succeq xx^\top$ approximated as $X \bullet UU^\top \geq \|\|U^\top x\|\|^2$ + affine constraints
LP	local marginal constraint $\sum_a X_{ia,jb} = x_{jb}$ and $x \in \Omega_s$
TRW	attempts to solve the dual of LP

Table 9.1: Summary of the convex relaxations presented in this report, which we classified in two categories: QP relaxations, and lift-and-project relaxations.

	# variables	# constraints	add. bound	mult. bound	higher order?
QP relax.					
L2QP	$\|V\|k$	$\|V\|$		$\frac{1}{k}$	yes
spectral	$\|V\|k$	$\|V\|$		$\frac{1}{k}$ (if $x \geq 0$)	no
CQP	$\|V\|k$	$\|V\|k$	$\frac{1}{4}\|\|W\|\|_1$		no
lift&project					
LP	$\|V\|k + \|E\|k^2$	$\|V\|k + \|E\|k^2$	$\frac{1}{4}\|\|W\|\|_1$	$\leq \frac{1}{k}$ for $k = 2$	yes
SOCP	$\|V\|k + \|E\|k^2$	$\|V\|k + \|E\|k^2$	$\frac{1}{4}\|\|W\|\|_1$		yes
SDP	$(\|V\|k)^2$	$(\|V\|k)^2$			yes

Table 9.2: Comparison of convex relaxations presented in this report. For each relaxation, we report the number of variables it uses as a function of the number of vertices $|V|$, edges $|E|$ and labels k, as well as the number of constraints, and potential additive or multiplicative bounds we can prove. We also report whether the relaxation naturally extends to higher order cliques.

- (3) $LOCAL(G) = MARG(G)$ when G is a tree, and there are non-tree cases where $\epsilon^*_{LOCAL(G)} = \epsilon^*_{MARG(G)}$ (implying tree agreement condition)
- (4) $LOCAL(G) \subset SOCP(G)$, *i.e.*the local SOCP constraints are *implied* by the local marginalization constraints despite the flexibility of the SOCP constraints
- (5) CQP is *equivalent* to $SOCP{-}LOCAL_1(G)$, in the sense that maximizing $\epsilon(x, X)$ over X obeying the SOCP constraints yields $\epsilon_{CQP}(x)$
- (6) the Lagrangian dual of the tree relaxation upper bound coincides with the LP relaxation on $LOCAL(G)$.

We have shown certain additive and multiplicative bounds for the different relaxations, see table 9.2. We also summarize the pros and cons of each relaxation in table 9.3. There are again a few surprising points:

	pros	cons
QP relax.	# variables, efficient solvers	performs poorly for larger k
L2QP	generalizes to $p - ary$ cliques	requires $W, V \geq 0$ (can affect bounds)
spectral	generalizable to tighten bounds	bounds only hold when $x \geq 0$
CQP	simple to implement	reduces to an SOCP
lift&project	generalizes to $p - ary$ cliques	# variables increases
SDP	very good approximation	inefficient, small-scale problems
SOCP	faster and sparser than SDP	most constraints *implied* by LP !
LP	optimal for trees, submodular, etc.	scales to mid-sized problems
TRW	very scalable	weak convergence guarantees

Table 9.3: Sampled pros and cons of each relaxation.

- (1) all additive bounds are the same despite strict dominance relationship between $LP - LOCAL_2(G)$ and $SOCP - LOCAL_1(G)$.
- (2) the multiplicative bounds for spectral relaxation and $L2QP$ are the same, and so are the data-dependant multiplicative bounds

9.2 Open questions

There is currently no dominance relation between $L2QP$, spectral relaxation and LP, but there are specific instances for which LP is optimal and not the other 2. It is unknown whether the same multiplicative bound applies for LP but we can probably show it cannot be better than $\frac{1}{k}$ for $k = 2$. The LP relaxation tends to give completely uniform fractional solutions for hard instances, which is not very informative. Is there an advantage in using one of the QP formulations in that case?

Another question relates to the generality of the techniques presented here. To what extent are they applicable to other problems such as QAP (Quadratic Assignment Problem) or other combinatorial problems? Is there a more general principle behind the different relaxations presented here that can potentially lead to tighter relaxations? We are currently investigating this with a novel formulation in terms of Lagrangian dual for a set of redundant quadratic constraints (such as $x_\alpha^2 = x_\alpha$ among other ones); we have shown in particular that this formulation *reduces* to most of the relaxations presented here (CQP, L2QP, spectral relaxation, SDP, LP), depending on the choice of redundant quadratic constraint that is used. We have also characterized a basis for the set of redundant quadratic *equality* constraints and determined its dimension for the MRF and QAP problems as a quotient space.

A more general question concerns whether or not the MRF formulation is the right model for the typical applications in computer vision and machine learning. For example in stereo reconstruction there is some evidence that the ground truth labeling often has a worse energy than the optimum for disparity estimation. This leads to the problem of designing the right cost functions for a specific task and may involve regularization, normalization[30] and/or learning[19],

Bibliography

[1] H. Ishikawa. Exact optimization for markov random fields with convex priors, 2003.

[2] Vladimir Kolmogorov and Ramin Zabih. What energy functions can be minimized via graph cuts? In *European Conference on Computer Vision*, 2002.

[3] Kevin P. Murphy, Yair Weiss, and Michael I. Jordan. Loopy belief propagation for approximate inference: An empirical study. pages 467–475.

[4] Weiss and Freeman. On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs. *IEEETIT: IEEE Transactions on Information Theory*, 47, 2001.

[5] M.J. Wainwright, T.S. Jaakkola, and A.S. Willsky. Map estimation via agreement on (hyper)trees: Message-passing and linear-programming approaches. 51(11):3697–3717, November 2005.

[6] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. In *ICCV (1)*, pages 377–384, 1999.

[7] Laurent Baratchart, Marc Berthod, and Loïc Pottier. Optimization of positive generalized polynomials under lp constraints. Technical Report RR-2750.

[8] Marius Leordeanu and Martial Hebert. Efficient map approximation for dense energy functions. In *International Conference on Machine Learning 2006*, May 2006.

[9] Pradeep Ravikumar and John Lafferty. Quadratic programming relaxations for metric labeling and markov random field map estimation. In *International Conference on Machine Learning*, 2006.

[10] Timothee Cour and Jianbo Shi. Solving markov random fields with spectral relaxation. In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics*, volume 11, 2007.

[11] P.H.S. Torr. Solving markov random fields using semidefinite programming. *aistats*, 2003.

[12] Masakazu Muramatsu and Tsunehiro Suzuki. A new second-order cone programming relaxation for max-cut problems. *Journal of Operations Research of Japan*, 46:164–177, 2001.

[13] M.Pawan Kumar, P.H.S. Torr, and A. Zisserman. Solving markov random fields using second order cone programming relaxations. *cvpr*, 1:1045–1052, 2006.

[14] Chandra Chekuri, Sanjeev Khanna, Joseph (Seffi) Naor, and Leonid Zosin. Approximation algorithms for the metric labeling problem via a new linear programming formulation. In *SODA '01: Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms*, pages 109–118, Philadelphia, PA, USA, 2001. Society for Industrial and Applied Mathematics.

[15] Arie M.C.A. Koster, van Hoesel, Stan P.M., and Antoon W.J. Kolen. The partial constraint satisfaction problem : facets and lifting theorems. Technical report, 1997.

[16] Jonathan S. Yedidia, William T. Freeman, and Yair Weiss. Understanding belief propagation and its generalizations. pages 239–269, 2003.

[17] Srinivas M. Aji, Gavin B. Horn, and Robert J. Mceliece. On the convergence of iterative decoding on graphs with a single cycle. In *In Proc. 1998 ISIT*, page 276, 1998.

[18] Amir Globerson and Tommi S. Jaakkola. Approximate inference using planar graph decomposition. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 473–480. MIT Press, Cambridge, MA, 2007.

[19] Ben Taskar, Vassil Chatalbashev, and Daphne Koller. Learning associative markov networks. In *Proc. ICML*. ACM Press, 2004.

[20] Julian E. Besag. On the statistical analysis of dirty pictures. In *Journal Of The Royal Statistical Society*, 1986.

[21] Timothee Cour and Ben Taskar. Tighter quadratic relaxations of energy functions via eigenvalue optimization. In *In preparation for AISTATS 2009*.

[22] N. Komodakis, N. Paragios, and G. Tziritas. Mrf optimization via dual decomposition: Message-passing revisited. pages 1–8, 2007.

[23] Owe Axelsson and Vincent A. Barker. *Finite element solution of boundary value problems: theory and computation*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2001.

[24] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. In *International Conference on Computer Vision*, 1999.

[25] Pawan Mudigonda, Vladimir Kolmogorov, and Philip Torr. An analysis of convex relaxations for map estimation. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1041–1048. MIT Press, Cambridge, MA, 2008.

[26] Pushmeet Kohli, Philip H. S. Torr, and Senior Member. Dynamic graph cuts for efficient inference in markov random fields.

[27] Nikos Komodakis, Georgios Tziritas, and Nikos Paragios. Performance vs computational efficiency for optimizing single and dynamic mrfs: Setting the state of the art with primal dual strategies. 2008.

[28] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient belief propagation for early vision. In *In CVPR*, pages 261–268, 2004.

[29] Vladimir Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1568–1583, October 2006.

[30] Timothee Cour, Praveen Srinivasan, and Jianbo Shi. Balanced graph matching. In B. Scholkopf, J.C. Platt, and T. Hofmann, editors, *Advances in Neural Information Processing Systems 19*. MIT Press, Cambridge, MA, 2007.