This CVPR2014 paper is the Open Access version, provided by the Computer Vision Foundation.
The authoritative version of this paper is available in IEEE Xplore.

Partial Optimality by Pruning for MAP-inference with General Graphical Models

Paul Swoboda[1], Bogdan Savchynskyy[2], Jörg H. Kappes[1], Christoph Schnörr[1,2]
[1]IPA,[2]HCI at Heidelberg University, Germany
{swoboda,kappes,schnoerr}@math.uni-heidelberg.de, bogdan.savchynskyy@iwr.uni-heidelberg.de

Abstract

We consider the energy minimization problem for undirected graphical models, also known as MAP-inference problem for Markov random fields which is NP-hard in general. We propose a novel polynomial time algorithm to obtain a part of its optimal non-relaxed integral *solution. Our algorithm is initialized with variables taking integral values in the solution of a convex relaxation of the MAP-inference problem and iteratively prunes those, which do not satisfy our criterion for partial optimality. We show that our pruning strategy is in a certain sense theoretically optimal. Also empirically our method outperforms previous approaches in terms of the number of persistently labelled variables. The method is very general, as it is applicable to models with arbitrary factors of an arbitrary order and can employ any solver for the considered relaxed problem. Our method's runtime is determined by the runtime of the convex relaxation solver for the MAP-inference problem.*

1. Introduction

Finding the most likely configuration of a Markov random field (MRF), also called MAP-inference or energy minimization problem for graphical models, is of big importance in computer vision, bioinformatics, communication theory, statistical physics, combinatorial optimization, signal processing, information retrieval and statistical machine learning, see [1, 10, 29] for an overview of applications. This key problem however is NP-hard. Therefore approximate methods have been developed to tackle big instances commonly arising in image processing, see [10, 27] for an overview of such methods. These approximate methods often cannot find an optimal configuration, but deliver close solutions. If one could prove, that some variables of the solution given by such approximate algorithms belong to an optimal configuration, the value of such approximate methods would be greatly enhanced. In particular, the problem for the remaining variables could be solved by stronger, but computationally more expensive methods to obtain a global optimum as done e.g. in [12].

In this paper we propose a way to gain such a partially optimal solution for the MAP-inference problem with *general* discrete MRFs from possibly also non-exact solutions of the commonly used local polytope relaxation (see [30]). Solving over the local polytope amounts to solving a linear problem for which *any* LP-solver can be used and for which dedicated and efficient algorithms exist.

1.1. Related Work

We distinguish two classes of approaches to partial optimality.

(i) Roof duality based approaches. The earliest paper dealing with persistency is [18], which states a persistency criterion for the stable set problem and verifies it for every solution of a certain relaxation. This relaxation is the same, as used by the roof duality method in [2] and which is also the basis for the well known QPBO-algorithm [2, 19]. The MQPBO method [14] extends roof duality to the multi-label case. The authors transform multi-label problems into quadratic binary ones and solve them via QPBO [2]. However, their transformation is dependent upon choosing a label order and their results are so as well, see the experiments in [26], where the label order is sampled randomly. It is not known how to choose an optimal label order to obtain the maximum number of persistent variables.

The roof duality method has been extended to higher order binary problems in [4, 7, 9]. The generalized roof duality method for binary higher order problems [9] computes partially optimal variables directly for higher order potentials, while Ishikawa's and Fix et al's approaches [4, 7] transform the higher order problem to one with unary and pairwise terms only. Fix et al's method [4] is an improvement upon Ishikawa's [7].

Windheuser et al [31] proposed *a multi-label higher-order* roof duality method, which is a generalization of both MQPBO [14] to higher order and Kahl and Strandmark's work [9] to the multi-label case. However Windheuser et al neither describe an implementation nor provide experimental validation for the higher order multi-label case.

(ii) Labeling testing approaches. A different approach, specialized for Potts models, is pursued by Kovtun [17], where possible labelings are tested for persistency by auxiliary submodular problems. The dead-end elimination procedure [3] tests, if certain labels of nodes cannot belong to an optimal solution. It is a local heuristic and does not perform any optimization.

Since for non-binary multi-labeling problems the submodular approximations constructed by approaches of class (i) are provably less tight than the standard local polytope relaxation [23, Prop. 1], we consider class (ii) in this paper. Specifically, based on ideas in [26] to handle the Potts model, we develop a theoretically substantiated approach to recognizing partial optimality for *general* graphical models, together with a competitive comparison to the 5 approaches [4, 7, 9, 14, 17] discussed above, that define the state-of-the-art.

Shrinking technique. The recent work [21] proposes a method for efficient shrinking of the combinatorial search area with the local polytope relaxation. Though the algorithmic idea is similar to the presented one, the method [21] does not provide partially optimal solutions. We refer to Section 4 for further discussion.

1.2. Contribution and Organization

Adopting ideas from [26], we propose *a novel* method for computing partial optimality, which is applicable to *general graphical models with arbitrary higher order potentials* and provides a higher number of persistent variables than the competing methods [4, 7, 9, 14, 17]. Similarly to [26] our algorithm is initialized with variables taking integral values in the solution of a convex relaxation of the MAP-inference problem and iteratively prunes those, which do not satisfy our persistency criterion. We show that our pruning strategy is in a certain sense theoretically optimal. Though the used relaxation can be chosen arbitrarily, for brevity we restrict our exposition and experiments to the local polytope one. Tighter relaxations *provably* yield better results. However even by using the local polytope relaxation we can achieve *a substantially higher* number of persistent variables, than competing approaches, which we confirm experimentally. Our approach is very general, as it can use *any*, also approximate, solver for the considered convex relaxation. Moreover, the computational complexity of our method is determined only by the runtime of the used solver.

The code is available at `http://paulswoboda.net`.

Organization. In Section 2 we review the energy minimization problem and the local polytope relaxation, in Section 3 we present our persistency criterion. The corresponding algorithm and its theoretical analysis are presented in Sections 4 and 5 respectively. Extensions to the higher order case and tighter relaxations are discussed in Section 6. Section 7 provides experimental validation of our approach and a comparison to the existing methods [4, 7, 9, 14, 17].

Due to the space limit the proofs were moved to the supplementary materials.

2. MAP-Inference Problem

The MAP-inference problem for a graphical model over an undirected graph $G = (\mathcal{V}, \mathcal{E})$, $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ reads

$$\min_{x \in X_{\mathcal{V}}} E_{\mathcal{V}}(x) := \sum_{v \in \mathcal{V}} \theta_v(x_v) + \sum_{uv \in \mathcal{E}} \theta_{uv}(x_u, x_v)\,, \quad (2.1)$$

where x_u belongs to a finite *label set* X_u for each node $u \in \mathcal{V}$, $\theta_u : X_u \to \mathbb{R}$ and $\theta_{uv} : X_u \times X_v \to \mathbb{R}$ are the *unary* and *pairwise potentials* associated with the nodes and edges of G. The label space for $A \subset \mathcal{V}$ is $X_A = \bigotimes_{u \in A} X_u$, where $\bigotimes$ stands for the Cartesian product. For notational convenience we write $X_{uv} = X_u \times X_v$ and $x_{uv} = (x_u, x_v)$ for $uv \in \mathcal{E}$. Notations like $x^0 \in X_A$ implicitly indicate that the vector x^0 only has components x^0_u indexed by $u \in A$. For two sets $A \subseteq B \subseteq \mathcal{V}$ and $x \in X_B$ denote by $x_{|A} \in X_A$ the restriction of x to A.

More general graphical models with terms depending on three or more variables can be considered as well. For brevity we restrict ourselves here to the pairwise case. An extension to the higher order case is discussed in Section 6.

Problem (2.1) is equivalent to the integer linear problem

$$\begin{aligned} &\min_{\mu \in \Lambda_{\mathcal{V}}} \sum_{v \in \mathcal{V}} \sum_{x_v \in X_v} \theta_v(x_v)\mu_v(x_v) + \sum_{uv \in \mathcal{E}} \sum_{x_{uv} \in X_{uv}} \theta_{uv}(x_{uv})\mu_{uv}(x_{uv}) \\ &\text{s.t.} \quad \mu_w(x_w) \in \{0,1\} \text{ for } w \in \mathcal{V} \cup \mathcal{E},\ x_w \in X_w\,, \end{aligned} \quad (2.2)$$

where *the local polytope* $\Lambda_{\mathcal{V}}$ [29] is the set of μ fulfilling

$$\begin{aligned} &\textstyle\sum_{x_v \in \mathcal{V}} \mu_v(x_v) = 1,\ v \in \mathcal{V}, \\ &\textstyle\sum_{x_v \in \mathcal{V}} \mu_{uv}(x_u, x_v) = \mu_u(x_u),\ x_u \in X_u,\ uv \in \mathcal{E}, \\ &\textstyle\sum_{x_u \in \mathcal{V}} \mu_{uv}(x_u, x_v) = \mu_v(x_v),\ x_v \in X_v,\ uv \in \mathcal{E}, \\ &\mu_{uv}(x_u, x_v) \geq 0,\ (x_u, x_v) \in X_{uv},\ uv \in \mathcal{E}\,. \end{aligned} \quad (2.3)$$

We define Λ_A for $A \subset \mathcal{V}$ similarly. Slightly abusing notation we will denote the objective function in (2.2)

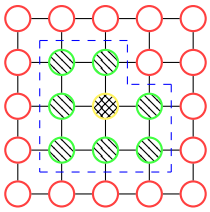

Figure 1. An exemplary graph containing inside nodes (yellow with crosshatch pattern) and boundary nodes (green with diagonal pattern). The blue dashed line encloses the set A. Boundary edges are those crossed by the dashed line.

as $E_{\mathcal{V}}(\mu)$. The formulation (2.2) utilizes the overcomplete representation [29] of labelings in terms of indicator vectors μ, which are often called *marginals*.The problem of finding $\mu^* \in \operatorname{argmin}_{\mu\in\Lambda_{\mathcal{V}}} E_{\mathcal{V}}(\mu)$ without integrality constraints is called *the local polytope relaxation* of (2.1).

While solving the local polytope relaxation can be done in polynomial time, the corresponding optimal marginal μ^* may not be integral anymore, hence infeasible and not optimal for (2.2). For a wide spectrum of problems however most of the entries of optimal marginals μ^* for the local polytope relaxation will be integral. Unfortunately, there is no guarantee that any of these integral variables will be part of a globally optimal solution to (2.2), except in the case of binary variables, that is $X_u = \{0,1\}\ \forall u \in \mathcal{V}$, and unary and pairwise potentials [5]. Natural questions are: (i) Is there a subset $A \subset V$ and a minimizer μ^0 of the original NP-hard problem (2.2) such that $\mu^0_v = \mu^*_v\ \forall v \in A$? In other words, is μ^* *partially optimal* or *persistent* on some set A? (ii) Given a relaxed solution $\mu^* \in \Lambda_{\mathcal{V}}$, how can we determine such a set A? We provide a novel approach to tackle these problems in what follows.

3. Persistency

Assume we have marginals $\mu \in \Lambda_A$ for $A \subseteq \mathcal{V}$. We say that the marginal μ_u is integral if $\mu_u(x_u) \in \{0,1\}$ $\forall x_u \in X_u, u \in A$. In this case the marginal corresponds uniquely to a label x_u with $\mu_u(x_u) = 1$.

Let the boundary nodes and edges of a subset of nodes $A \subset \mathcal{V}$ be defined as follows:

Definition 1 (Boundary and Interior)**.** *For the set $A \subseteq V$ the set $\partial\mathcal{V}_A := \{u \in A : \exists v \in \mathcal{V}\backslash A \text{ s.t. } uv \in \mathcal{E}\}$ is called* its boundary*. The respective* set of boundary edges *is defined as $\partial\mathcal{E}_A = \{uv \in \mathcal{E} : u \in A \text{ and } v \in \mathcal{V}\backslash A\}$. The set $A\backslash\partial\mathcal{V}_A$ is called the interior of A.*

An exemplary graph illustrating the concept of interior and boundary nodes can be seen in Figure 1.

Definition 2 (Persistency)**.** *A labeling $x^0 \in X_A$ on a subset $A \subset V$ is* partially optimal *or* persistent *if x^0 coincides with an optimal solution to (2.1) on A.*

In the remainder of this section, we state our novel persistency criterion in Theorem 1. Taking additionally into account convex relaxation yields a computationally tractable approach in Corollary 1.

As a starting point, consider the following sufficient criterion for persistency of $x^0 \in X_A$. Introducing a *concatenation* of labelings $x^0 \in X_A$ and $\tilde{x} \in X_{V\backslash A}$ as $(x^0, \tilde{x}) := \begin{cases} x^0_v, & v \in A, \\ \tilde{x}_v, & v \in \mathcal{V}\backslash A \end{cases}$, the criterion reads:

$$\forall \tilde{x} \in X_{V\backslash A} : \ (x^0, \tilde{x}) \in \operatorname*{argmin}_{\substack{x\in X_{\mathcal{V}} \\ x_v = \tilde{x}_v \forall v \in \mathcal{V}\backslash A}} E_{\mathcal{V}}(x)\,. \tag{3.1}$$

This means that if we fix *any* labeling $\tilde{x}$ on the complement of A and optimize with respect to x^0 on A, the concatenated labeling $(x^0, \tilde{x})$ will be always optimal. Informally this means that the solution x^0 is independent of what happens on $V\backslash A$. This criterion however is hard to check directly, as it entails solving NP-hard minimization problems over an exponential number of labelings $\tilde{x} \in X_{\mathcal{V}\backslash A}$.

We relax the above criterion (3.1) so that we have to check the solution of only *one* energy minimization problem by modifying the unaries θ_v on boundary nodes so that they bound the influence of *all* labelings on $V\backslash A$ uniformly.

Definition 3 (Boundary potentials and energies)**.** *For a set $A \subset V$ and a boundary labeling $y \in X_{\partial\mathcal{V}_A}$, we define for each boundary edge $uv \in \partial\mathcal{E}_A$, $u \in \partial\mathcal{V}_A$ the "boundary" potential $\hat{\theta}_{uv,y_u} : X_u \to \mathbb{R}$ as follows:*

$$\hat{\theta}_{uv,y_u}(x_u) := \begin{cases} \max_{x_v\in X_v} \theta_{uv}(x_u, x_v), & y_u = x_u \\ \min_{x_v\in X_v} \theta_{uv}(x_u, x_v), & y_u \neq x_u \end{cases}. \tag{3.2}$$

Define the energy $\hat{E}_{A,y}$ on A with boundary labeling y as

$$\hat{E}_{A,y}(x) := E_A(x) + \sum_{uv\in\partial\mathcal{E}_A\,:\,u\in\partial\mathcal{V}_A} \hat{\theta}_{uv,y_u}(x_u)\,, \tag{3.3}$$

where $E_A(x) = \sum_{u\in A} \theta_u(x_u) + \sum_{uv\in\mathcal{E}:u,v\in A} \theta_{uv}(x_{uv})$ is the energy with potentials in A.

Given a boundary labeling $y \in \partial\mathcal{V}_A$, we have worsened the energy in (3.3) for all labelings conforming to y and made them more favourable for all labelings not conforming to y. An illustration of a boundary potential can be found in Figure 2. As a consequence, if there is an optimal solution to the energy (3.3) which is equal to the boundary labeling y on $\partial\mathcal{V}_A$, Theorem 1 below shows that it is not affected by what happens outside A and is hence persistent on A.

Theorem 1 (Partial optimality criterion)**.** *A labeling $x^0 \in X_A$ on a subset $A \subset V$ is persistent if*

$$x^0 \in \operatorname{argmin}_{x\in X_A} \hat{E}_{A,x^0_{|\partial\mathcal{V}_A}}(x)\,. \tag{3.4}$$

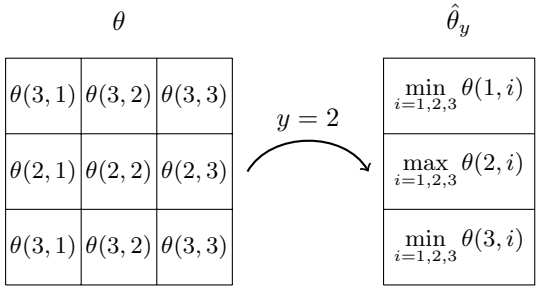

Figure 2. Illustration of a boundary potential $\hat{\theta}_y$ constructed in (3.2). The second label comes from the boundary conditions y, therefore entries are maximized for the second row and minimized otherwise.

Checking the criterion in Theorem 1 is NP-hard, because (3.4) is a MAP-inference problem of the same class as (2.1). By relaxing the minimization problem (3.4) one obtains the polynomially verifiable persistency criterion in Corollary 1.

Corollary 1 (Tractable partial optimality criterion)**.** *Suppose marginals $\mu^0 \in \Lambda_A$ and a labeling $x^0 \in X_A$ are given such that $\mu^0_u(x^0_u) = 1 \ \forall u \in A$ (in particular μ^0_u, $u \in A$, is integral). If*

$$\mu^0 \in \operatorname{argmin}_{\mu \in \Lambda_A} \hat{E}_{A, x^0_{|\partial \mathcal{V}_A}}(\mu)\,, \tag{3.5}$$

x^0 is persistent on A.

4. Persistency Algorithm

Now we concentrate on finding a set A and labeling $x \in X_A$ such that the solution of $\min_{\mu \in \Lambda_A} \hat{E}_{A, x_{|\partial \mathcal{V}_A}}(\mu)$ fulfills the conditions of Corollary 1. Our approach is summarized in Algorithm 1.

In the initialization step of Algorithm 1 we solve the relaxed problem over $\mathcal{V}$ without boundary labeling and initialize the set A^0 with nodes having an integer label. Then in each iteration t we minimize over the local polytope the energy $\hat{E}_{A^t, x^t_{|\partial \mathcal{V}_{A^t}}}$ defined in (3.3), corresponding to the set A^t and boundary labeling coming from the solution of the last iteration. We remove from A^t all variables which are not integral or do not conform to the boundary labeling. In each iteration t of Algorithm 1 we shrink the set A^t by removing variables taking non-integral values or not conforming to the current boundary condition.

Convergence. Since $\mathcal{V}$ is finite and $|A^t|$ is monotonically decreasing, the algorithm converges in at most $|\mathcal{V}|$ steps. Solving each subproblem in Algorithm 1 can be done in polynomial time. As the number of iterations of Algorithm 1 is at most $|\mathcal{V}|$, Algorithm 1 itself is polynomial as well. In practice only few iterations are needed.

Algorithm 1: Finding persistent variables.

Data: $G = (\mathcal{V}, \mathcal{E})$, $\theta_u : X_u \to \mathbb{R}$, $\theta_{uv} : X_{uv} \to \mathbb{R}$
Result: $A^* \subset \mathcal{V}$, $x^* \in X_{A^*}$
Initialize:
Choose $\mu^0 \in \operatorname{argmin}_{\mu \in \Lambda_\mathcal{V}} E_\mathcal{V}(\mu)$
$A^0 = \{u \in \mathcal{V} \colon \mu^0_u \in \{0,1\}^{|X_u|}\}$
$t = 0$
repeat
  Set x^t_u such that $\mu^t_u(x^t_u) = 1, u \in \mathcal{V}_{A^t}$
  Choose $\mu^{t+1} \in \operatorname{argmin}_{\mu \in \Lambda_{A^t}} \hat{E}_{A^t, x^t_{|\partial \mathcal{V}_{A^t}}}(\mu)$
  $t = t + 1$
  $W^t = \{u \in \partial \mathcal{V}_{A^{t-1}} \ : \ \mu^t_u(x^{t-1}_u) \neq 1\}$
  $A^t = \{u \in A^{t-1} \colon \mu^t_u \in \{0,1\}^{|X_u|}\} \backslash W^t$
until $A^t = A^{t-1}$;
$A^* = A^t$
Set $x^* \in X_{A^*}$ such that $\mu^t_u(x^*_u) = 1$

After termination of Algorithm 1, we have

$$\mu^* \in \operatorname{argmin}_{\mu \in \Lambda_{A^*}} \hat{E}_{A^*, x^*_{|\partial \mathcal{V}_{A^*}}}(\mu)\,, \tag{4.1}$$

μ^* is integral and μ^* and x^* correspond to the same labeling on $\partial \mathcal{V}_A$. Hence μ^*, x^* and A^* fulfill the conditions of Corollary 1, which proves persistency.

Choice of Solver. All our results are independent of the specific algorithm one uses to solve the relaxed problems $\min_{\mu \in \Lambda_A} \hat{E}_{A,y}$, provided it returns an exact solution. However this can be an issue for large-scale datasets, where classical exact LP solvers like e.g. the simplex method become inapplicable. It is important that one can also employ *approximate* solvers, as soon as they provide (i) a proposal for *potentially* persistent nodes and (ii) sufficient conditions for optimality of the found *integral* solutions such as e.g. zero duality gap. These properties have the following precise formulation.

Definition 4 (Consistent labeling)**.** *A labeling $c \in \bigotimes_{v \in \mathcal{V}} (X_v \cup \{\#\})$ is called* a consistent labeling *for the energy minimization problem (2.1), if from $c_v \in X_v$ $\forall v \in \mathcal{V}$ follows that $c \in \operatorname{argmin}_{x \in X} E_\mathcal{V}(x)$.*

We will call an algorithm for solving the energy minimization problem (2.1) consistency ascertaining*, if it provides a consistent labeling as its output.*

Consistent labelings can be constructed for a wide range of algorithms, e.g.:

- Dual decomposition based algorithms [11, 15, 16, 20, 22] deliver *strong tree agreement* [28] and algorithms considering the Lagrange dual return *strong arc consistency* [30] for some nodes. If for a node one of these properties holds, we set c_v as the corresponding label. Otherwise we set $c_v = \#$.

- Naturally, any algorithm solving $\min_{\mu\in\Lambda_\mathcal{V}} E(\mu)$ exactly is consistency ascertaining with $c_v = \begin{cases} x_v, & \mu_v(x_v) = 1 \\ \#, & \mu_v \notin \{0,1\}^{|X_v|} \end{cases}$.

Proposition 1. *Let operators $\mu \in \operatorname{argmin}(...)$ in Algorithm 1 be exchanged with*

$$\forall v \in \mathcal{V}, x_v \in X_v,\ \mu_v(x_v) := \begin{cases} 1, & c_v = x_v \\ 0, & c_v \notin \{x_v, \#\}, \\ 1/|X_v|, & c_v = \# \end{cases}$$

where c are consistent labelings returned by a consistency ascertaining algorithm applied to the corresponding minimization problems. Then the output labeling x^* is persistent.

Comparison to the Shrinking Technique [21]. The recently published approach [21], similar to Algorithm 1, describes how to shrink the combinatorial search area with the local polytope relaxation. However (i) Algorithm 1 solves a series of auxiliary problems on the subsets A^t of integer labels, whereas the method [21] considers nodes, which got fractional labels in the relaxed solution; (ii) Algorithm 1 is polynomial and provides only persistent labels, whereas the method [21] has exponential complexity and either finds an optimal solution or gives no information about persistence. While the subsets of variables to which the method [21] applies a combinatorial solver to achieve global optimality are often smaller than those of our present method, because potentials remain unchanged in contrast to the perturbation (3.3), the presented methods finds the largest persistent labeling with regard to the persistency criterion in Corollary 1 as detailed next.

5. Largest Persistent Labeling

Let $A^0 \subseteq V$ and $\mu^0 \in \Lambda_{A^0}$ be defined as in Algorithm 1. Subsets $A \subset A^0$ which fulfill the conditions of Corollary 1 taken with labelings $\mu^0|_A$ can be partially ordered with respect to inclusion $\subset$ of their domains. In this section we will show that the following holds:

- There is a largest set among those, for which there exists a *unique* persistent labeling fufilling the conditions of Corollary 1.
- Algorithm 1 finds this largest set.

This will imply that Algorithm 1 cannot be improved upon with regard to the criterion in Corollary 1.

Definition 5 (Strong Persistency)**.** *A labeling $x^* \in X_A$ is called* strongly persistent *on A, if x^* is the* unique *labeling on A fulfilling the conditions of Theorem 1.*

Theorem 2 (Largest persistent labeling)**.** *Algorithm 1 finds a superset A^* of the largest set $A^*_{strong} \subseteq A^* \subset V$ of strongly persistent variables identifiable by the criterion in Corollary 1.*

Corollary 2. *If there is a* unique *solution of $\min_{\mu\in\Lambda_{A^t}} \hat{E}_{A^t, x^t_{\partial\mathcal{V}_A}}(\mu)$ for all $t = 0, \dots$ obtained during the iterations of Algorithm 1, then Algorithm 1 finds the largest subset of persistent variables identifiable by the sufficient partial optimality criterion in Corollary 1.*

If uniqueness of the optimal marginals μ^t during the execution of Algorithm 1 does not hold, then Algorithm 1 is not deterministic and the obtained set A^* is not necessarily the largest persistent set identifiable by the criterion in Corollary 1. The simplest example of such a situation occurs if the relaxation $\min_{\mu\in\Lambda_\mathcal{V}} E_\mathcal{V}(\mu)$ is tight, but has several integer solutions. Any convex combination of these solutions will form a non-integral solution. However this fact cannot be recognized by our method and hence the non-integral variables of the solution will be discarded.

6. Extensions

Higher Order. Assume now we are not in the pairwise case anymore but have an energy minimization problem over a *hyper*graph $G = (\mathcal{V}, \mathcal{E})$ with $\mathcal{E} \subset \mathcal{P}(\mathcal{V})$ a set of subsets of $\mathcal{V}$:

$$\min_{x\in X} E_\mathcal{V}(x) := \sum_{e\in\mathcal{E}} \theta_e(x_e)\,. \tag{6.1}$$

All definitions, our persistency criterion and Algorithm 1 admit a straightforward generalization. Analoguously to Definition 1 define for a subset of nodes $A \subset \mathcal{V}$ the boundary nodes as

$$\partial\mathcal{V}_A := \{u \in A\colon \exists v \in \mathcal{V}\backslash A, \exists e \in \mathcal{E} \text{ s.t. } u, v \in e\} \tag{6.2}$$

and the boundary edges as

$$\partial\mathcal{E}_A := \{e \in \mathcal{E}\colon \exists u \in A, \exists v \in \mathcal{V}\backslash A \text{ s.t. } u, v \in e\}\,. \tag{6.3}$$

The equivalent of boundary potential in Definition 3 for $e \in \partial\mathcal{E}_A$ is

$$\hat{\theta}_{e,y}(x) := \begin{cases} \max\limits_{\tilde{x}\in X_e\colon \tilde{x}_{|A\cap e} = x_{|A\cap e}} \theta_e(\tilde{x}), & x_{|A\cap e} = y_{|A\cap e} \\ \min\limits_{\tilde{x}\in X_e\colon \tilde{x}_{|A\cap e} = x_{|A\cap e}} \theta_e(\tilde{x}), & x_{|A\cap e} \neq y_{|A\cap e} \end{cases}. \tag{6.4}$$

Now Theorem 1, Corollary 1 and Algorithm 1 can be directly translated to the higher order case.

Tighter Relaxations. Essentially, Algorithm 1 can be applied also to tighter relaxations than Λ_A, e.g.

when one includes cycle inequalities [24]. One merely has to replace the local polytope Λ_A for $A \subset V$ by the tighter feasible convex set:

Proposition 2. *Let the polytope $\tilde{\Lambda}_A$ contain all integral marginals of Λ_A and be such that $\tilde{\Lambda}_A \subset \Lambda_A$ $\forall A \subseteq \mathcal{V}$. Use $\tilde{\Lambda}_A$ in place of Λ_A in Algorithm 1 and let $\tilde{A}^*$ be the corresponding persistent set returned by the modified algorithm. Let $A^*_{strong} \subseteq A^*$ be the largest subset of strongly persistent variables identifiable by Corollary 1 subject to the relaxations $\tilde{\Lambda}_A$ and Λ_A. Then $A^*_{strong} \subseteq \tilde{A}^*_{strong}$.*

7. Experiments

We tested our approach on several datasets from different computer vision and machine learning benchmarks, 96 problem instances overall, see Table 1. We describe each dataset and corresponding experiments in detail below.

Competing methods. We compared our method to **MQPBO** [14,23], **Kovtun**'s method [17], Generalized Roof Duality (**GRD**) by Kahl and Strandmark [9], Fix et al's [4] and Ishikawa's Higer Order Clique Reduction (**HOCR**) [7] algorithms. For the first two methods we used our own implementation, and for the other the freely available code of Strandmark [25]. We were unable to compare to the method of Windheuser et al. [31], because the authors do not give a description for implementing their method in the higher order case and only provide experimental evaluation for problems with pairwise potentials, where their method coincides with MQPBO [14].

Implementation details. We employed TRWS as an approximate solver for Algorithm 1 and strong tree agreement as a consistency mapping (see Proposition 1) for most of the pairwise problems. We stop TRWS once it has either arrived at tree-agreement, a small duality gap of 10^{-5} or 1000 iterations. For the `side-chain` pairwise models and all higher-order models we employed CPLEX as an exact linear programming solver, because TRWS either was not applicable or got stuck in clearly suboptimal points.

Datasets and Evaluation. We give a brief characterization of all datasets and report the obtained total percentage of persistent variables of our and competing methods in Table 1. We refer to the supplementary material for detailed results for each individual problem instance.

The problem instances `teddy`, `venus`, `family`, `pano`, `Potts` and `geo-surf` were made available by [10], while the datasets `side-chain` and `protein-interaction` were made available by [1].

The problem instances `teddy` and `venus` come from

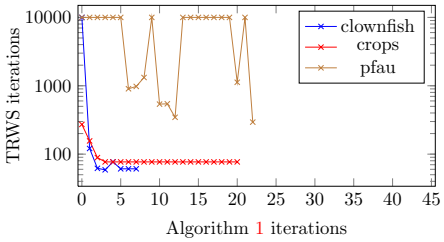

Figure 3. Iterations needed by TRWS [15] in Algorithm 1 for three `Potts` instances.

the disparity estimation for stereo vision [27]. None of the competing approaches was able to find even a single persistent variable for these datasets, presumably because of the large number of labels, whereas we labeled nearly half of them as persistent in `teddy`, though only 15 in `venus`.

Instances named `pano` and `family` come from the **photomontage** dataset [27]. These problems have more complicated pairwise potentials than the disparity estimation problems, but less labels. For both datasets we found significantly more persistent variables than MQPBO, in particular, we were able to label more than half of the variables in `pano`.

We also chose 12 relatively big energy minimization problems with grid structure and **Potts** interaction terms. The underlying application is a color segmentation problem previously considered in [26]. Our general approach reproduces results of [26] for the specific Potts model. See detailed results in our supplementary materials.

We considered also `side-chain` prediction problems in **protein folding** [32]. The datasets consist of pairwise graphical models with $32 - 1971$ variables and $2 - 483$ labels. The problems with fewer variables are densely connected and have very big label spaces, while the larger ones are less densely connected and have label space up to ~ 100 variables. Our method labels as persistent *an order of magnitude more* nodes than MQPBO.

The `protein interaction` models [8] aim to find the subset of proteins, which interact with each other. Our method labeled more than three quarter of nodes as persistent, whereas all methods absed on roof duality, i.e. Fix et at, GRD, HOCR [4, 7, 9], gave similar results and labeled around a quarter of them as persistent.

The **cell tracking** problem consists of a binary higher order graphical model [13]. Given a sequence of microscopy images of a growing organism, the aim is to find the lineage tree of all cells, which divide themselves and move. This is done by tracking developing cells across time. For implementation reasons

Experiment	#I	#L	#V	O	MQPBO	Kovtun	GRD	Fix	HOCR	Ours
teddy	1	60	168749	2	0	†	†	†	†	**0.4423**
venus	1	20	166221	2	0	†	†	†	†	**0.0009**
family	1	5	425631	2	0.0432	†	†	†	†	**0.0611**
pano	1	7	514079	2	0.1247	†	†	†	†	**0.5680**
Potts	12	≤12	≤424720	2	0.1839	0.7475	†	†	†	**0.9231**
side-chain	21	≤483	≤1971	2	0.0247	†	†	†	†	**0.6513**
protein-interaction	8	2	≤14440	3	†	†	0.2603	0.2545	0.2545	**0.7799**
cell-tracking	1	2	41134	9	†	†	†	0.1771	†	**0.9992**
geo-surf	50	7	≤1111	3	†	†	†	†	†	**0.8407**

Table 1. Percentage of persistent variables obtained by methods [4, 7, 9, 14, 17] and our method. Notation † means inapplicability of the method. The columns #I,#L,#V,O denote the number of instances, labels, variables and the highest order of potentials respectively. We refer to the supplementary material for results for each individual dataset. The column "Ours" reveals the superior performance of our approach.

we were not able to solve `cell-tracking` dataset with Ishikawa's [7] method. However Fix [4] reports that his method outperforms Ishikawa's method [7]. Other methods are not applicable even theoretically, whereas we managed to label as persistent more than 99.9% of the nodes.

Last, we took the higher order multi-label **geometric surface labeling problems** (denoted as `geo-surf` in Table 1) from [6]. They consist of instances with $29 - 1111$ variables and 7 labels each with unary, pairwise and ternary terms. Note that MQPBO cannot handle ternary terms, Fix et al's [4] Ishikawa's [7] methods and the generalized roof duality method by Strandmark and Kahl [9] cannot handle more than 2 labels. Hence we report our results without comparison. We considered only those 50 instances out of the total 300, which could not be solved with the local polytope relaxation. Again the number of variables, which we were able to mark as persistent is high - more than 80% on average.

Runtime. The runtime of our algorithm mainly depends on the speed of the underlying solver for the local polytope relaxation. Currently there seems to be no general rule regarding the runtime of our algorithm. We show three iteration counts for instances of the `Potts` dataset in Figure 3. In the `clownfish` instance the number of iterations of TRWS [15] drops fast after the first iteration. On the `crops` instance the number of iterations is initially much lower, however it does only decrease moderately and more iterations are needed to prune variables. For the hard `pfau` instance Algorithm 1 needed many iterations and number of TRWS [15] iterations does not drop significantly. We refer to the supplementary materials for more plots.

Pruning. In all our experiments, set A^0 in Algorithm 1 contained at least 97% (but usually more than 99%) of the variables, hence at most 3% of all variables were pruned initially due to not being consistent. In subsequent rounds always more than 99.99% of all variables were consistent, and variables were mainly pruned due to not agreeing with the boundary conditions.

8. Conclusion and Outlook

We have presented a novel method for finding persistent variables for undirected graphical models. Empirically it outperforms all tested approaches with respect to the number of persistent variables found on every single dataset. Our method is general: it can be applied to graphical models of arbitrary order and type of potentials. Moreover, there is no fixed choice of convex relaxation for the energy minimization problem and arbitrary, also approximate, solvers for these relaxations can be employed in our approach.

In the future we plan to significantly speed-up the implementation of our method and consider finer persistency criteria, which are also able to ascertain persistency not only in terms of variables taking a single label but falling into a range of labels.

Acknowledgments. This work has been supported by the German Research Foundation (DFG) within the program "Spatio-/Temporal Graphical Models and Applications in Image Analysis", grant GRK 1653. The authors also thank Marco Esquinazi for helpful discussions.

References

[1] The probabilistic inference challenge (PIC2011). http://www.cs.huji.ac.il/project/PASCAL/. 1, 6

[2] E. Boros and P. L. Hammer. Pseudo-Boolean optimization. *Discrete Applied Mathematics*, 2002. 1

[3] J. Desmet, M. D. Maeyer, B. Hazes, and I. Lasters. The dead-end elimination theorem and its use in protein side-chain positioning. *Nature*, 356, 1992. 2

[4] A. Fix, A. Gruber, E. Boros, and R. Zabih. A graph cut algorithm for higher-order Markov random fields. In *ICCV*, 2011. 1, 2, 6, 7

[5] P. Hammer, P. Hansen, and B. Simeone. Roof duality, complementation and persistency in quadratic 0-1 optimization. *Math. Programming*, 28, 1984. 3

[6] D. Hoiem, A. A. Efros, and M. Hebert. Recovering surface layout from an image. *IJCV*, 75(1), 2007. 7

[7] H. Ishikawa. Transformation of general binary MRF minimization to the first-order case. *PAMI*, 33(6), June 2011. 1, 2, 6, 7

[8] A. Jaimovich, G. Elidan, H. Margalit, and N. Friedman. Towards an integrated protein-protein interaction network: A relational Markov network approach. *Jour. of Comp. Biol.*, 13(2), 2006. 6

[9] F. Kahl and P. Strandmark. Generalized roof duality. *Discrete Applied Mathematics*, 160(16-17), 2012. 1, 2, 6, 7

[10] J. H. Kappes, B. Andres, F. A. Hamprecht, C. Schnörr, S. Nowozin, D. Batra, S. Kim, B. X. Kausler, J. Lellmann, N. Komodakis, and C. Rother. A comparative study of modern inference techniques for discrete energy minimization problem. In *CVPR*, 2013. 1, 6

[11] J. H. Kappes, B. Savchynskyy, and C. Schnörr. A bundle approach to efficient MAP-inference by Lagrangian relaxation. In *CVPR*, 2012. 4

[12] J. H. Kappes, M. Speth, G. Reinelt, and C. Schnörr. Towards efficient and exact MAP-inference for large scale discrete computer vision problems via combinatorial optimization. In *CVPR*, 2013. 1

[13] B. X. Kausler, M. Schiegg, B. Andres, M. S. Lindner, U. Köthe, H. Leitte, J. Wittbrodt, L. Hufnagel, and F. A. Hamprecht. A discrete chain graph model for 3d+t cell tracking with high misdetection robustness. In *ECCV*, 2012. 6

[14] P. Kohli, A. Shekhovtsov, C. Rother, V. Kolmogorov, and P. Torr. On partial optimality in multi-label MRFs. In *ICML*, 2008. 1, 2, 6, 7

[15] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *PAMI*, 28(10), Oct. 2006. 4, 6, 7

[16] N. Komodakis, N. Paragios, and G. Tziritas. MRF energy minimization and beyond via dual decomposition. *PAMI*, 33(3), 2011. 4

[17] I. Kovtun. Partial optimal labeling search for a NP-hard subclass of (max,+) problems. In *DAGM*, 2003. 2, 6, 7

[18] G. L. Nemhauser and L. E. Trotter. Vertex packings: Structural properties and algorithms. *Math. Programming*, 8, 1975. 1

[19] C. Rother, V. Kolmogorov, V. S. Lempitsky, and M. Szummer. Optimizing binary MRFs via extended roof duality. In *CVPR*, 2007. 1

[20] B. Savchynskyy, J. Kappes, S. Schmidt, and C. Schnörr. A study of Nesterov's scheme for Lagrangian decomposition and MAP labeling. In *CVPR*, 2011. 4

[21] B. Savchynskyy, J. H. Kappes, P. Swoboda, and C. Schnörr. Global MAP-optimality by shrinking the combinatorial search area with convex relaxation. In *NIPS*, 2013. 2, 5

[22] B. Savchynskyy, S. Schmidt, J. H. Kappes, and C. Schnörr. Efficient MRF energy minimization via adaptive diminishing smoothing. In *UAI*, 2012. 4

[23] A. Shekhovtsov, V. Kolmogorov, P. Kohli, V. Hlavac, C. Rother, and P. Torr. LP-relaxation of binarized energy minimization. Technical Report CTU–CMP–2007–27, Czech Technical University, 2008. 2, 6

[24] D. Sontag. *Approximate Inference in Graphical Models using LP Relaxations*. PhD thesis, Massachusetts Institute of Technology, 2010. 6

[25] P. Strandmark. Generalized roof duality. http://www.maths.lth.se/matematiklth/personal/petter/pseudoboolean.php. 6

[26] P. Swoboda, B. Savchynskyy, J. H. Kappes, and C. Schnörr. Partial optimality via iterative pruning for the Potts model. In *SSVM*, 2013. 1, 2, 6

[27] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. F. Tappen, and C. Rother. A comparative study of energy minimization methods for Markov random fields with smoothness-based priors. *PAMI*, 30(6), 2008. 1, 6

[28] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. MAP estimation via agreement on trees: message-passing and linear programming. *IEEE Trans. Inf. Theor.*, 51(11), 2005. 4

[29] M. J. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.*, 1(1-2):1–305, Jan. 2008. 1, 2, 3

[30] T. Werner. A linear programming approach to max-sum problem: A review. *PAMI*, 29(7), 2007. 1, 4

[31] T. Windheuser, H. Ishikawa, and D. Cremers. Generalized roof duality for multi-label optimization: Optimal lower bounds and persistency. In *ECCV*, 2012. 2, 6

[32] C. Yanover, O. Schueler-Furman, and Y. Weiss. Minimizing and learning energy functions for side-chain prediction. *Jour. of Comp. Biol.*, 15(7), 2008. 6