CANADA EXCELLENCE RESEARCH CHAIR

DATA SCIENCE FOR REAL-TIME DECISION-MAKING

AN ILP SOLVER FOR MULTI-LABEL MRFS WITH CONNECTIVITY CONSTRAINTS

Ruobing Shen
Eric Kendinibilir
Ismail Ben Ayed
Andrea Lodi
Andrea Tramontani
Gerhard Reinelt

December 2017

DS4DM-2017-016

POLYTECHNIQUE MONTRÉAL
DÉPARTEMENT DE MATHÉMATIQUES ET GÉNIE INDUSTRIEL
Pavillon André-Aisenstadt
Succursale Centre-Ville C.P. 6079
Montréal - Québec
H3C 3A7 - Canada
Téléphone: 514-340-5121 # 3314

An ILP Solver for Multi-label MRFS with Connectivity Constraints

Ruobing Shen*[1], Eric Kendinibilir[2], Ismail Ben Ayed[3], Andrea Lodi[4], Andrea Tramontani[5], and Gerhard Reinelt[1]

[1]*Institue of Computer Science, Heidelberg University, Germany*
[2]*Institute of Mathematics, Heidelberg University, Germany*
[3]*LIVIA Laboratory, École de technologie supérieure (ETS), Montreal, QC, Canada*
[4]*Department of Mathematical and Industrial Engineering, Polytechnique Montréal, Canada*
[5]*CPLEX Optimization, IBM, Italy*

Abstract

Integer Linear Programming (ILP) formulations of Markov random fields (MRFs) models with global connectivity priors were investigated previously in computer vision, e.g., [16, 17]. In these works, only Linear Programming (LP) relaxations [16, 17] or simplified versions [21] of the problem were solved. This paper investigates the ILP of multi-label MRF with exact connectivity priors via a branch-and-cut method, which provably finds globally optimal solutions. The method enforces connectivity priors iteratively by a cutting plane method, and provides feasible solutions with a guarantee on sub-optimality even if we terminate it earlier. The proposed ILP can also be applied as a post-processing method on top of any existing multi-label segmentation approach. We demonstrate the power and usefulness of our model by several experiments on the BSDS500 image dataset, as well as on medical images with trained probability maps.

1. Introduction

Most early vision problems can be formulated using Markov Random Fields (MRFs), hence its solution algorithms are of pivotal importance in computer vision. The MAP-MRF (maximizing a posteriori in an MRF) has proven to be successful for many computer vision problems such as image segmentation, denoising and stereo, among others. We refer to [9, 10, 20] for an overview of MRF optimization techniques and applications in vision.

In the standard case of MRF with pairwise potentials, we have an undirected graph $G = (V, E)$, where V represents a set of pixels (or superpixels) from an input image, and E denotes a set of edges consisting of unordered pairs of nodes indicating adjacency relations. We consider the problem of minimizing the following energy function:

$$E(x) = \sum\nolimits_{p\in V} \theta_p(x_p) + \sum\nolimits_{(p,q)\in E} \theta_{pq}(x_p, x_q). \quad (1)$$

Here, we use x_p to denote the label of node $p \in V$, which belongs to a pre-defined finite set $\mathcal{L} = [k]$ representing k classes, where $[k] = \{1, \dots, k\}$. $\theta_p(x_p)$ is usually called unary potential, and is derived from the observed data. It measures how well label x_p fits node p. $V_{pq}(x_p, x_q)$ is often referred to as pairwise potential. It measures the cost of assigning labels x_p, x_q to adjacent nodes p, q. Typically, it is used to impose spatial smoothness or to align the solution boundaries to image edges. The goal is to find a labeling x (i.e., a mapping from V to $\mathcal{L}$) that minimizes $E(x)$. The Potts function $\theta(\alpha, \beta) = \lambda \cdot \mathbb{1}(\alpha \neq \beta)$, where λ is a constant, and $\mathbb{1}(\cdot)$ is 1 if its argument is true and 0 otherwise, is widely used, among many other models.

Minimizing energy (1) is a difficult problem (NP-hard in general). In the case of an undirected graph, and by introducing binary variables x_i^ℓ, $i \in V$, $\ell \in \mathcal{L}$, which indicate whether node i is assigned label ℓ ($x_i^\ell = 1$ in this case), the corresponding ILP formulation with Potts function boils down to:

$$\min_x \ (1-\lambda) \sum_{\ell=1}^{k} \sum_{i=1}^{n} c_i^\ell x_i^\ell + \lambda \sum_{\ell=1}^{k} \sum_{(i,j)\in E} |x_i^\ell - x_j^\ell| \quad (2)$$

$$\sum\nolimits_{\ell=1}^{k} x_i^\ell = 1, \quad \forall i \in [n], \quad (2a)$$

$$x_i^\ell \in \{0,1\}, \quad \forall i \in [n], \quad \ell \in [k], \quad (2b)$$

*Electronic address: ruobing.shen@informatik.uni-heidelberg.de

where c_i^ℓ denotes the unary data term for label ℓ and node i, and $\lambda \in [0, 1]$ is a positive constant weighting the contribution of the smoothness term. Constraint (2a) enforces that each node is assigned exactly one label.

Since (2) is NP-hard and difficult to solve to optimality, it is common in vision to solve the corresponding LP relaxation [12, 13]. There have been works on solving approximations of (2), for instance, message passing algorithms [11, 22] and α-expansion [4] with guaranteed approximation ratios. The corresponding condition is nonnegative edge weights and $V_{pq}(\beta, \gamma) + V_{pq}(\alpha, \alpha) \leq V_{pq}(\beta, \alpha) + V_{pq}(\alpha, \gamma)$, for all labels $\alpha, \beta, \gamma \in \mathcal{L}$.

The standard model in (2), which combines unary and pairwise potentials, can impose only a limited class of constraints on the solution. Therefore, there is an ongoing research effort in computer vision towards embedding high-order constraints in MRFs. These includes, for instance, region connectivity [16, 17, 21], shape convexity [8], curvature regularization [15] and shape compactness [6], among other high-order priors. In this paper, we investigate exact region connectedness priors. More precisely, we are interested in solving (2) to global optimality, while adding a global (high-order) potential function $C(x)$ to (2) to explicitly enforce the connectivity of each segment (to be made more precise in Sec. 2). A *k-label partitioning* of the image in this paper is a partition of V into connected subsets $\{C_1, C_2, \ldots, C_k\}$ such that $\cup_{i=1}^k C_i = V$, and $C_i \cap C_j = \emptyset$, $i \neq j$. Without loss of generality, we assume that segment C_i is assigned the label i. Enforcing the connectivity potential itself is proven to be NP-hard in [21].

1.1. Related Works

Image segmentation under approximate connectivity constraints has been considered in [21], where a binary MRF is solved. Exact connectivity is not considered in [21]. Instead, a simplified version of the problem is proposed, where only a given (user-provided) pair of nodes must be connected. Following this assumption, the problem is solved with a heuristic-based graph cut algorithm [3], obtaining connected foreground (binary segmentation).

Exact global connectivity potentials are formulated as an ILP in [17], where connected subgraph polytopes are introduced. Due to the high computational cost of solving the corresponding NP-hard problem, the work in [17] examined only LP relaxations of the ensuing ILP. Although the general formulation works for multi-label MRFs, the authors applied it only to binary MRF problems. In [18], the authors optimized a linear (unary-potential) objective subject to connectivity constraint in a binary (two-region) partitioning problem. The model does not apply to the general multi-label pairwise MRF objective in (1), which is of wide interest in vision applications. Finally, it is worth mentioning that the subgraph connectivity problem also plays an important role in the operations research community, and has been applied, for instance, to the forest planning problem [5], where each subregion of the forest is constrained to be connected.

1.2. Contribution

This paper investigates multi-label MRFs with exact connectivity constraints. To solve the ensuing ILP problem, we propose a branch-and-cut method, which provably finds globally optimal solutions. The method enforces connectivity priors iteratively by a cutting plane method, and provides feasible solutions with a guarantee on suboptimality even if we terminate it earlier. Unlike [16, 17], which examines LP relaxations of the initial ILP, our method provides global optimality guarantee. Different from [21], we consider exact connectivity and we do not reduce the problem to connectivity between a given pair of points. The proposed ILP is quite general, and can also be applied as a post-processing method on top of any existing multi-label segmentation approach.

2. Connected Subgraphs

In this section we introduce the set of all connected subgraphs, where a connected subgraph is just a set of nodes with the same label that are connected.

Connected subgraph. Given a connected, undirected graph $G = (V, E)$, we call $C_\ell = G_\ell(V_\ell, E_\ell)$ a connected subgraph with label ℓ if G_ℓ is connected, where $V_\ell = \{i \in V : x_i^\ell = 1\}$ and $E_\ell = \{(i, j) \in E : i, j \in V_\ell\}$. Recall that a subgraph $G'(V', E')$ is connected if $\forall i, j \in V'$, $\exists$ a path in G' that connects i and j. We call a node $i \in V$ *active* in V_ℓ if $x_i^\ell = 1$, i.e., if it has label ℓ.

Vertex-Separator Set. Given a subgraph $G_\ell = (V_\ell, E_\ell)$, for any pair of active nodes $i, j \in V_\ell$, $i \neq j$, $(i, j) \notin E_\ell$, the set $S_\ell \subseteq V \setminus \{i, j\}$ is said to be a vertex-separator set with respect to $\{i, j\}$ if the removal of S_ℓ from G disconnects i and j in G. It also means that there exists no path between i and j in $G' = (V \setminus S_\ell, E \setminus (S_\ell \times V))$. As an additional definition, a set $\bar{S}_\ell$ is said to be a *minimal vertex-separator set* in label ℓ if it is a vertex-separator set with respect to a pair $\{i, j\}$ of active nodes in V_ℓ and any strict subset $T \subset \bar{S}_\ell$ is not.

Let $\mathcal{S}_\ell(i, j) = \{S_\ell \subset V : S_\ell$ is a vertex-separator with respect to active node pair $\{i, j\}$ in $V_\ell\}$ be the collection of all $\{i, j\}$ vertex-separator sets, and $\bar{\mathcal{S}}_\ell(i, j) \subset \mathcal{S}_\ell(i, j)$ be the subsets of minimal vertex-separator sets.

Following [17], we can describe C_ℓ with the following set of linear inequalities:

$$x_i^\ell + x_j^\ell - 1 \leq \sum\nolimits_{s \in S_\ell} x_s^\ell, \quad \forall i, j \in V : (i, j) \notin E,\ \forall S_\ell \in \mathcal{S}_\ell(i, j), \tag{3}$$

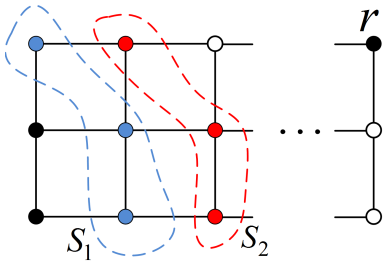


Figure 1: *K-Nearest cut generation strategy*. Active nodes are shown in black, and the two separator sets are marked in red and blue.

where $x_i^\ell \in \{0,1\}$, $i \in V$ and $\ell \in [k]$. In other words, if two nodes i and j are active in S_ℓ (left hand side of (3) becomes 1), they are not allowed to be separated by any set of inactive nodes of S_ℓ.

In [17], the authors also prove inequalities (3) are facet-defining for the convex hull of C_ℓ if $\mathcal{S}_\ell(i,j)$ is replaced by $\bar{\mathcal{S}}_\ell(i,j)$.

Rooted case. In this paper, we require the user to input a scribble for each label, so that at least one node is identified within each label. Let r_ℓ denotes the root node for label ℓ. We use the first node of the scribble as the root node. Then, it suffices to check connectivity of every active node to the root node instead of all pairs of active nodes associated with the label. Thus, constraints (3) become

$$x_i^\ell \leq \sum\nolimits_{s\in S_\ell} x_s^\ell, \quad \forall i \in V : (i, r_\ell) \notin E, \; \forall S_\ell \in \mathcal{S}_\ell(i, r_\ell). \tag{4}$$

In practice, the number of constraints (4) is exponentially large for any label ℓ, hence they cannot be considered all simultaneously for graphs of large sizes. However, given a labeling x^ℓ, we can identify a subset of violated connectivity constraints in polynomial time and iteratively add them to the ILP while searching for new integer solutions. This is known as the *cut generation* approach. We will look into this in detail in Sec. 4.1.

3. MRFs with Connectivity Constraints

3.1. Proposed model: ILP-PC

Let f_i denotes the observed image feature (e.g., color) at spatial location i. We assume the user inputs k scribbles as seeds for the k labels, as shown in the left image of Fig. 3. Assuming image observations follow a piecewise constant model within each region[1], let Y_ℓ denotes the image average of seeds within label ℓ. In this case, unary potential $c_i^\ell = |f_i - Y_\ell|$ evaluates how well label ℓ fits node i.

[1]We assume a piecewise constant model for simplicity. However, our formulation extends to any other probabilistic assumptions of observation models.

By introducing two nonnegative variables $\varepsilon_i^{\ell+}$ and $\varepsilon_i^{\ell-}$ to model $|x_i^\ell - x_j^\ell|$, the ILP of our multi-label MRF with connectivity constraints becomes:

$$\min_x (1-\lambda) \sum_{\ell=1}^{k} \sum_{i=1}^{n} c_i^\ell x_i^\ell + \lambda \sum_{\ell=1}^{k} \sum_{(i,j)\in E} (\varepsilon_i^{\ell+} + \varepsilon_i^{\ell-}) \tag{5}$$

$$\sum\nolimits_{\ell=1}^{k} x_i^\ell = 1, \quad \forall i \in [n], \; \ell \in [k], \tag{5a}$$

$$x_i^\ell - x_j^\ell = \varepsilon_i^{\ell+} - \varepsilon_i^{\ell-}, \quad \forall i \in [n], \; \ell \in [k], \tag{5b}$$

$$x_i^\ell \in \{0,1\}, \quad \forall i \in [n], \; \ell \in [k], \tag{5c}$$

$$x_i^\ell \in C_\ell, \quad \forall i \in [n], \; \ell \in [k], \tag{5d}$$

$$\varepsilon_i^{\ell+}, \varepsilon_i^{\ell-} \geq 0, \quad \forall i \in [n], \; \ell \in [k], \tag{5e}$$

$$x_i^\ell = 1, \quad \forall i \text{ within the scribble of label } \ell, \tag{5f}$$

where constraints (5d) can be expressed as the rooted vertex-separator constraints (4).

In the case of a superpixel graph, where a superpixel contains similar pixels in terms of color or texture, we represent relations between neighboring superpixels by defining the corresponding *Region Adjacency Graph* (RAG) $G = (V, E)$, where E contains edges between pairs of adjacent superpixels. We multiply the unary data term by σ_i and the pairwise term by γ_{ij}. Here, σ_i denotes the number of pixels contained in node (superpixel) i, and γ_{ij} represents the number of neighboring pixels between node i and j.

3.2. ILP-PCB: ILP-PC with background label

If a clear background (not necessarily connected) exists in the given image, the connectivity constraints can be ignored on the specific label, which we call the background label. This is a reasonable assumption in many cases, such as the black-region background in the left image of Fig. 4. In this example, the background has 4 disconnected components. Fig. 4 depicts results with and without background label.

3.3. ILP-PCO: ILP-PC with ordering constraints

It is common practice in integer programming to use redundant constraints in the formulation to speed up computation. A constraint is redundant if it is not strictly needed for a formulation to be valid, i.e., forbidding all integer infeasible solutions. However, redundant constraints can be useful because they forbid fractional solutions that could be encountered during the branch-and-bound approach, where one iteratively solves the LP relaxation, or because they impose a structure in the solution that, for example, breaks symmetries. The latter is the case of constraints

$$\sum\nolimits_{i\in V} x_i^\ell \geq \sum\nolimits_{i\in V} x_i^{\ell+1}, \quad \forall \ell \in [n-1], \tag{6}$$

where we assume that the number of pixels in segment ℓ is at least as large as that of segment $\ell+1$, thus imposing an

order of the segments that is generally beneficial (see Sec. 5) to enumerate symmetric solutions. Indeed, two solutions obtained by swapping labels in any pair of segments ℓ, k are perfectly equivalent without constraints (6).

4. Solution Techniques

4.1. Cut generation strategies

We are interested in exact connectivity and we focus on the rooted case (4), since we assume at least one root node r_ℓ is fixed for each label ℓ by the user interaction.

We concentrate on enforcing the connectivity constraints for one label only (e.g., ℓ). Then, the same approach will be repeated for other label sets until they are all connected (in the case of a background label, we could ignore its connectivity). The basic idea is to omit (5d) initially, explore the branch-and-bound tree of system (5) until an integer solution is found and then check the feasibility of this solution (i.e., connectivity of G_ℓ). If not feasible, violated constraints (5d) are separated and added to (5) to cut off the infeasible solution. This procedure is iterated until G_ℓ is connected.

We treat individual connected components (see Fig. 1) as one entity, since establishing connectivity between all nodes in this component and r_ℓ automatically connects all the nodes. Identifying violated constraints (5d) then boils down to finding a vertex separator set S_ℓ between each disconnected active component and the root component (containing r_ℓ) in the current solution.

At the heart of the above technique is that only a subset of connectivity constraints (5d) will be active at the optimum of (5). However, depending on the choice of the inequalities that we choose in each step, we may require a different number of such cuts and the number of iterations varies.

Among the many ways of separating and selecting the violated constraints (5d), we choose the so-called K-Nearest strategy. Namely, we run a breath-first search from any active component H to collect the K (disjoint) separator sets S_m, $m = 1, \ldots, K$ composed of all nodes with identical distance. The search terminates if k equals the number of nodes in H or if another active node is reached. The idea is illustrated in Figure 1, where active nodes are shown in black and r denotes the root node. The two separator sets are marked in red and blue. Here $K = 2$ since it reaches the number of nodes in H.

The K-Nearest strategy is reported in [18] to be one of the most successful (among five) in terms of solved instances and computational efficiency. We will adopt this strategy in Sec. 5.

4.2. L_0-H: a region fusion based heuristic

To improve the running time of our ILP solver, we calculate an initial feasible solution with a heuristic, called L_0-H, since it will be an upper bound to the branch-and-cut method the solver uses. This, on the one hand, helps to prune a lot of unnecessary branching nodes. On the other hand, the solver can provide an optimality gap to the initial solution, by solving the LP relaxation of the ILP, which will serve as a lower bound to the problem.

We adopted the idea for the heuristic from [14], which works by iteratively merging groups of nodes. In the beginning, each scribble of nodes and every node not covered by any scribbles are in their own groups. Then for every iteration, we merge two neighboring groups, if the following condition holds and the merging does not result in the nodes of two different scribbles being in the same group:

$$\sigma_i \cdot \sigma_j \cdot |Y_i - Y_j|^2 \leq \beta \cdot \gamma_{ij} \cdot (\sigma_i + \sigma_j). \qquad (7)$$

where σ_i denotes the number of pixels in group (segments) i, γ_{i_j} denotes the number of neighboring pixels (boundary length) of two groups i and j, and Y_i the mean of image data (e.g., color) within group i.

By increasing β in (7) in every iteration, we terminate the algorithm when exactly k groups remain. Let η denotes the growing rate, it is shown in [14] that the following exponentially growing strategy of β gives the best results.

$$\beta = (\frac{\text{iter}}{100})^{2.2} * \eta$$

where iter is the current iteration number. We will show in Sec. 5 that L_0-H is fast and generates good results most of the time, sometimes even optimal.

5. Experiments

In this paper, all computational experiments were performed using Cplex 12.7.0, on a Intel i5-4570 quad-core machine, with 16 GB RAM. We show experiments on medical images, where the unary potentials are based on the probability maps of given labels, which were trained using convolutional neural networks (CNN) [7]. The sizes range from 96×96 to 256×256.

We further use the Berkeley Segmentation Dataset [2] (BSDS500), which contains 500 natural images, each of size 321×481. We apply the SLIC [1] superpixel algorithm to get an over-segmentation, where the superpixel size is around 1000.

Using superpixels has several advantages. First, the complexity of the optimization problem is drastically reduced with only a negligible segmentation error. Second, the information in each superpixel is more discriminative, and also overcomes the case of outliers. As shown in a recent superpixel algorithms survey paper [19], a few

advanced superpixel algorithms can achieve very accurate over-segmentation results with around 1000 suerpixels.

We conduct a comprehensive comparison of the following different optimization models:

- ILP-PC. Our proposed ILP formulation (5) of multi-label MRF, under the global connectivity constraints.
- LP-PC: The LP relaxation of ILP-PC, which was introduced in [17, 18].
- L_0-H: Our proposed L_0 region fusion based heuristic, which was motivated by [14] and modified to generate exactly k connected segments.
- ILP-P: The ILP formulation of (5) without connectivity constraints (5d), which is widely used in vision (e.g., graph cuts).
- ILP-PCB: ILP-PC with the "background" label marked by the user, where this special label is not required to be connected.
- ILP-PCO: ILP-PC plus the ordering constraints (6).

In this section, if there is no further explanation, the default setting for the pairwise potential λ is 0.2, 100 sec for the time limit, and 0.1 for the L_0-H parameter η. When we report energy E, we use the objective function in (5).

5.1. Detailed Comparison

5.1.1 Medical images with probability maps

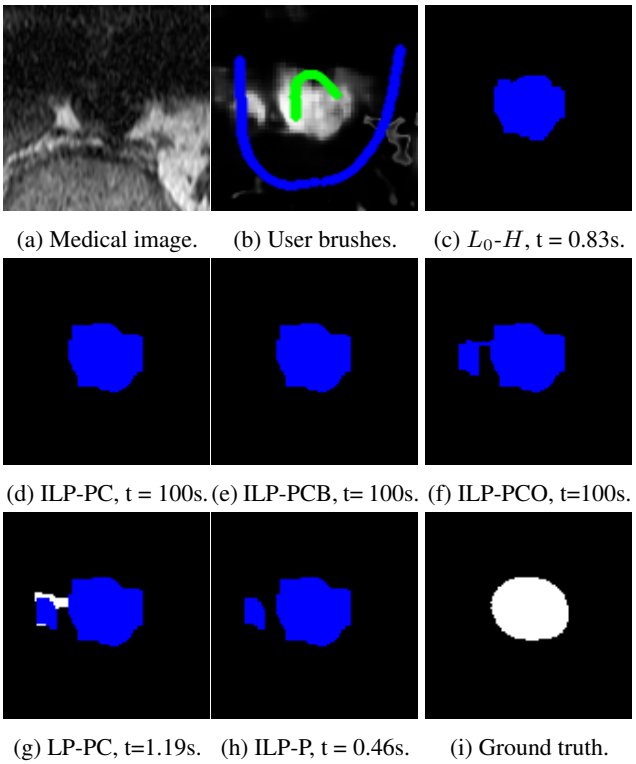
We report a medical image segmentation example, where unary potentials are based on the probability maps of given labels, which were trained using convolutional neural networks (CNN) [7]. The purpose here is to obtain a binary (two-region) segmentation of a magnetic resonance image (MRI), which depicts the abdominal aorta [6]. In this example, the CNN probability maps yielded unsatisfying disconnected region due to imaging noise, the lack of boundary contrast and limited training information.

The input image is of size 111×111, and the computation time is reported in Fig. 2. ILP-PC, ILP-PCB and ILP-PCO all failed to converge given the initial solution provided by L_0-H. L_0-H result is of high quality, within 1.22% of the best solution found by ILP-PC in 100 seconds. The energy is reported in the following table.

H	PC	PCB	PCO	LPC	P
864.5	854	854	839	829.5	826.8

For the sake of space, we have omitted all the prefixes of the models, and abbreviated LP-PC as LPC.

As we see in Fig. 2, LP-PC has $0,62\%$ fractional solution (depicted in white). Although a post-processing rounding heuristic can be applied, it is not guaranteed to find a feasible solution. ILP-P gives two separated regions, which is far away from the ground truth. We notice that LP-PC and ILP-P give lower energy than ILP-PC. This is because both of them are relaxations for ILP-PC and, therefore, provide lower bounds.



(a) Medical image. (b) User brushes. (c) L_0-H, t = 0.83s.
(d) ILP-PC, t = 100s. (e) ILP-PCB, t= 100s. (f) ILP-PCO, t=100s.
(g) LP-PC, t=1.19s. (h) ILP-P, t = 0.46s. (i) Ground truth.

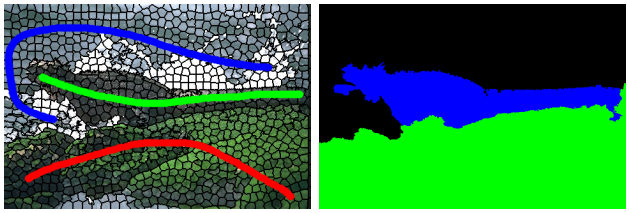
Figure 2: Comparison of 5 models on a medical image, where user scribbles are depicted on the probability map. Energy is reported in Sec. 5.1.1, and t denotes the time spent. In LP-PC, 1.22% of the pixels remains unlabeled , colored in white. Both ILP-PC and ILP-PCB have 2.8% gap, while ILP-PCO has 0.9% , with lower energy (not necessarily visually better).

The inclusion of background label is not beneficial in this example, while the ordering constraints (6) help find lower energy, and reduces the optimality gap from 2.8% to 0.9%. It is worth to mention that unwanted solutions like ILP-PCO in this example (although with lower energy) can be easily avoided by drawing slightly different scribbles on purpose.

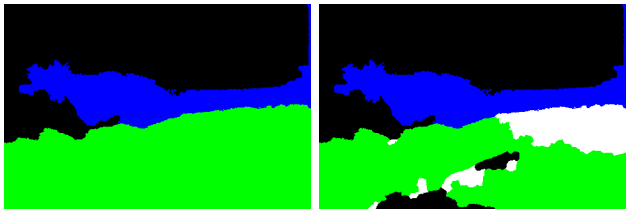
5.1.2 Superpixels of BSDS500

In this section, we introduce another model ILP-PCW, which is ILP-PC without the initial solution of L_0-H. The purpose is to test whether the ILP solver is able to achieve good results by itself.

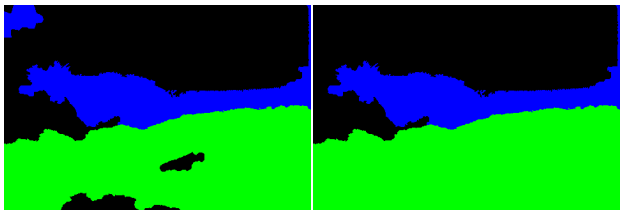
Fig. 3 depicts an example, where ILP-PC with L_0-H does not converge within the time limit while ILP-PCW finds the provably global optimal solution. The solution



(a) Input with 3 user scribbles. (b) L_0-H, t = 0.04s, E=16088.



(c) ILP-PC, t=100 s, E=15804.9. (d) LP-PC, t=0.27 s, E=15560.9.



(e) ILP-P, t=0.08s, E=15232.5. (f) ILP-PCW, t=61s, E=15804.9.

Figure 3: Comparison of 4 models on BSDS, plus ILP-PCW (ILP-PC without initial solution from L_0-H). t: time spent, E: energy. Note 5.9% of the nodes remains unlabeled in LP-PC, colored in white. The L_0-H solution is within 1.76% of the optimal.

time and the energy are reported in the figure. The energy of the starting solution provided by L_0-H is very good, within 1.76% of the optimal solution found by ILP-PCW in 61 seconds. Note that ILP-PC and ILP-PCW give the same energy, meaning they found the same solution, but ILP-PC failed to get the tightest lower bound, having an optimality gap of 0.3%. A closer look into the log file of Cplex shows that, given the good initial solution of L_0-H, ILP-PC found the "best solution" in less than 1 sec, while ILP-PCW found the same solution in 18 sec.

The inclusion of the integrality constraints and the connectivity priors greatly improve solution quality. As many as 5.9% superpixel values of LP-PC are factional. In ILP-P, the green and black labels have several disconnected regions, resulting in an worse solution.

5.1.3 More experiments on BSDS500 images

More experiments on BSDS500 images are shown in Fig. 6. In the first column, the pairwise term λ is set to 0.1 to encourage thin branches of the tree, while all other parameters remain at their default values. We draw much fewer brushes in the right two columns, to show the robustness of our model (to be discussed in Sec. 5.3 with more details). Note that the white pixels in LP-PC denote fractional solutions, and ILP-P is without connectivity constraints, thus allowing disconnected regions with the same label. We observe that L_0-H gives good results in the right two images, while not being satisfying in the left two cases. Our proposed model ILP-PC achieves the best overall results.

5.2. Quantitative Comparison

In this section, we give a detailed analysis of the different models with respect to energy, computational time and parameters. They are based on computational experiments of 15 images from BSDS500 and real medical images from [6].

5.2.1 Time, ILP gap and fractional solutions of LP-PC

We report the average running time (T) of all models in the second row of the table below, where the time limit is 100 sec. The average ILP optimality gap (G) is shown in the third row, where "Null" means no optimality gap exists (since they are not ILP). For the sake of space, we again omit all the prefixes of the models.

	H	PC	PCB	PCO	LPC	P
T	0.7	62.3	39.2	71.8	1.4	0.3
G	Null	3.7%	1.9%	2.1%	Null	0

We can clearly see that the ILP models often hit the 100 seconds time limit. However, the solution is within 4% of optimality. Moreover, the inclusion of the "background label" (ILP-PCB) and ordering constraints (ILP-PCB) helps in term of speed and optimal gap (with the exception that ILP-PCO takes more time). It is surprising to see that ILP-P with only pairwise priors is more efficient than L_0-H.

Apart from the above statistics, we also report that among all tested images an average of 3.5% pixels found by LP-PC remain unlabeled (they have fractional solutions).

5.2.2 ILP-PCB against ILP-PC.

Based on the computational experiments, we report the gain and loss on using ILP-PCB against ILP-PC, if a clear background exists in that image. The score is calculated as follows.

If both models solve the problem to optimality, we report the computational time difference. If both models failed to solve the problem within the time limit, we report the ILP optimality gap difference. In the cases where one model solves the problem to optimality while the other reaches the time limit, we report the computational time difference assuming that the latter takes 100 secs.

On 12 instances where the application of ILP-PCB makes sense, 8 instances benefit, 3 lose while one instance

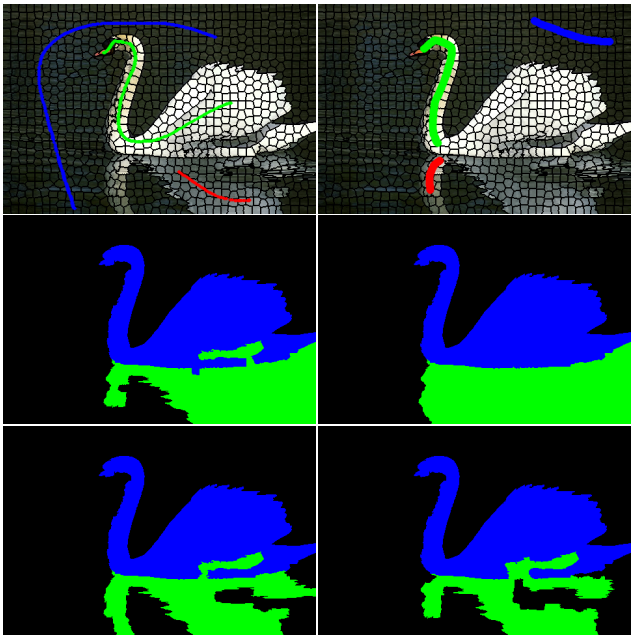


Figure 4: Different user scribbles on the same image. Second row: ILP-PC with E=9544.2 and 13422.8, both reaching 100 seconds time limit. Bottom row: ILP-PCB with E=8615.3 and 10482.3, t= 2.1s and 0.4s. Note that the background label (shown in black) can be disconnected, while the other labels (blue and green) are connected.

remains the same. The average net gain in computation time equals 28.6 sec, in which there exist two instances that reduce the time of 100 secs from ILP-PC to less than one sec. The average net gain in ILP optimality gap equals 0.9%. This important gain results from "relaxing" one label to be non-connected.

Apart from speed gain, it can be beneficial in practice to use ILP-PCB when a clear background with disconnected regions exists; See Fig. 4.

5.2.3 ILP-PCO against ILP-PC.

We use the same methodology as above. On 10 instances where the application of ordering constraints (6) (ILP-PCO) make sense, 7 instances benefit while 3 lose. The average net loss on computation time equals 1.2 secs, while the average net gain in ILP optimality gap equals 1.6%. While we do not observe a clear advantage of applying ILP-PCO, it often works better on hard instances (in which the ILP solver converges very slowly). It is worth to mention that in one instance where no obvious ordering exists, it took ILP-PCO 100 secs and still did not find any feasible solution.

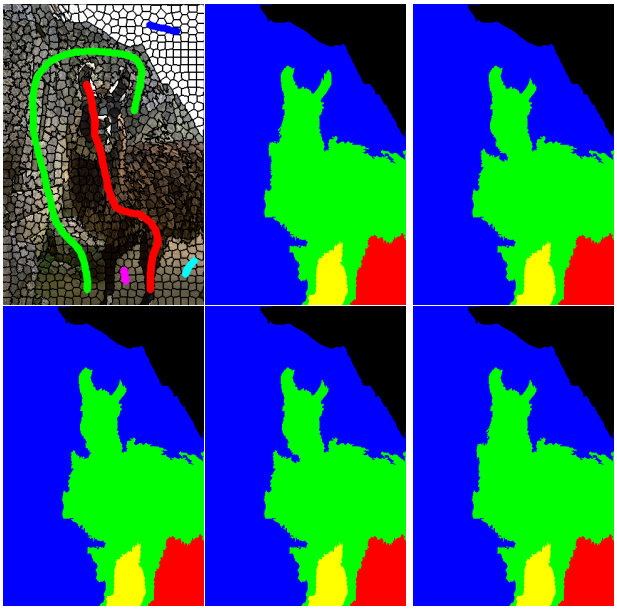


Figure 5: Experiments on L_0-H with different η. From top left to bottom right: parameter η changes from 0.1 to 0.7 (with interval 0.15). The time reduces from 0.08 to 0.05 secs.

5.3. Analysis of different user scribbles

The user scribbles are used to learn the average color of each label, which is used in the ILP as the unary term. They also enforce hard constraints into the ILP (5), which help fixing some of the binary variables, thus pruning the branch-and-bound search trees within the ILP solver. Moreover, in case of difficult situations, scribbles can also be used to exclude outliers from one label, such as in Fig.2. We show in Fig. 4 that changing the scribbles does not alter significantly the results. While ILP-PC reaches the time limit in both cases, ILP-PCB gets the reported optimal solution in only 2.1 and 0.4 secs. The energy differences between the two cases are due to two factors: different scribbles resulted not only into different hard constraints, but also different unary potentials.

5.4. Report and analysis on L_0-H

L_0-H is fast to compute, only taking 0.7 seconds on average to generate a feasible solution, which could be used as initialization for the ILP solver. Other than its efficiency, the quality in terms of energy is on average only 7.20% with respect to the best solution ILP-PC can find within 100 seconds time limit.

For all the other experiments reported in this paper, the growing rate parameter η is set to be 0.1. Fig. 5 shows an experiment on adopting different η. We use 5 different values of parameter η, ranging from 0.1 to 0.7, with an in-

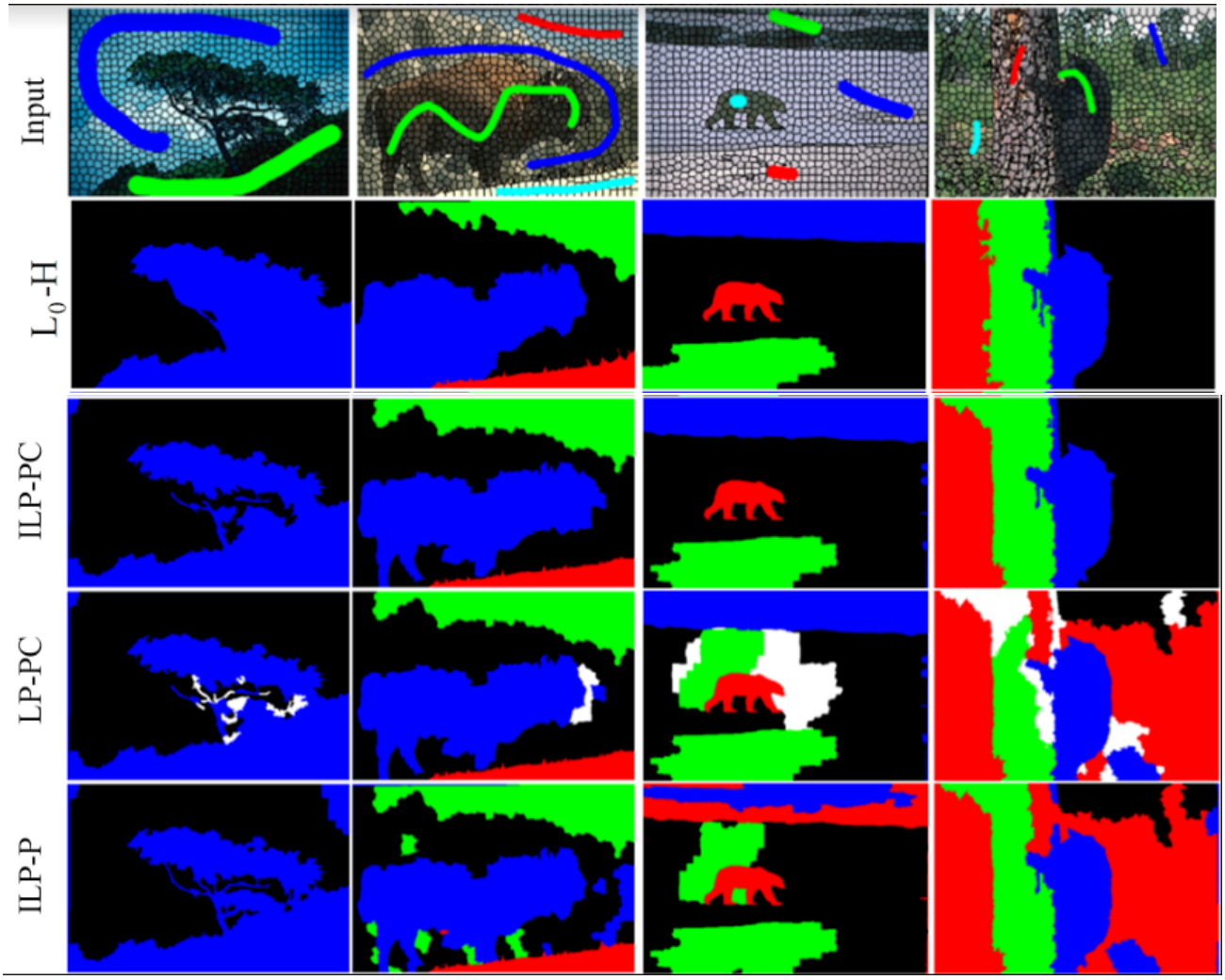


Figure 6: More experiments on BSDS500 images. The pairwise term λ is set to 0.1 to encourage thin branches of the tree in the first column, while all other parameters remain default. The user brushes in the right two columns are few on purpose, to show the robustness of our model. The white pixels in LP-PC denote fractional solutions, and ILP-P is without connectivity constraints, thus allowing disconnected regions with the same label.

terval of 0.15 between each pair of successive values. The computation time decreases with the increase of η, reducing gradually from 0.08 to 0.05 secs. One can observe that L_0-H is robust with respect to its parameter, with only slight changes in the results.

6. Conclusion

Recent years' algorithmic advances in Mixed Integer Programming (MIP) plus the hardware improvements have resulted in a enormous speedup in solving MIP. We revisit the ILP of the multi-label MRF with connectedness priors, and propose a cutting plane approach to exactly enforce the connectivity constraints. A fast region fusion based heuristic is designed to provide a good initial solution. The solver provides a feasible or better solution with a guarantee on the sub-optimality even if we terminate it earlier.

The ILP can also be applied as a post-processing method on top of any existing multi-label segmentation methods. Hence the advantage of ILP is two-fold. On the one hand, it provides a guarantee (lower bound) for any given initial solution. On the other hand, it seeks for better solutions during its search in the branch-and-bound tree.

In this paper, we demonstrated the power and usefulness of our model by some experiments in the BSDS500, and medical images with trained probability maps. We show that with moderate-sized images or superpixels of large ones, our model achieves the best overall performance, yielding provably global optimum in some instances.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282, Nov 2012. 4

[2] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(5):898–916, May 2011. 4

[3] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, Sep 2004. 2

[4] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, Nov 2001. 2

[5] R. Carvajal, M. Constantino, M. Goycoolea, J. P. Vielma, and A. Weintraub. Imposing connectivity constraints in forest planning models. *Operations Research*, page 824, Jul 2013. 2

[6] J. Dolz, I. B. Ayed, and C. Desrosiers. Unbiased shape compactness for segmentation. In *Medical Image Computing and Computer Assisted Intervention - MICCAI (1)*, pages 755–763, Sep 2017. 2, 5, 6

[7] J. Dolz, C. Desrosiers, and I. B. Ayed. 3d fully convolutional networks for subcortical segmentation in mri: A large-scale study. *NeuroImage*, Apr 2017. 4, 5

[8] L. Gorelick, O. Veksler, Y. Boykov, and C. Nieuwenhuis. Convexity shape prior for binary segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(2):258–271, Feb 2017. 2

[9] J. H. Kappes, B. Andres, F. A. Hamprecht, C. Schnörr, S. Nowozin, D. Batra, S. Kim, B. X. Kausler, T. Kröger, J. Lellmann, N. Komodakis, B. Savchynskyy, and C. Rother. A comparative study of modern inference techniques for structured discrete energy minimization problems. *International Journal of Computer Vision*, 115(2):155–184, Nov 2015. 1

[10] R. Kindermann and J. L. Snell. *Markov random fields and their applications*. Providence, R.I. : American Mathematical Society, 1980. References p.133-142. 1

[11] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1568–1583, Oct 2006. 2

[12] N. Komodakis, N. Paragios, and G. Tziritas. MRF energy minimization and beyond via dual decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(3):531–552, Mar 2011. 2

[13] N. Komodakis and G. Tziritas. Approximate labeling via graph cuts based on linear programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(8):1436–1453, Aug 2007. 2

[14] R. M. H. Nguyen and M. S. Brown. Fast and effective l0 gradient minimization by region fusion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 208–216, Dec 2015. 4, 5

[15] C. Nieuwenhuis, E. Toeppe, L. Gorelick, O. Veksler, and Y. Boykov. Efficient squared curvature. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 4098–4105, June 2014. 2

[16] S. Nowozin and C. Lampert. Global interactions in random field models: A potential function ensuring connectedness. *SIAM Journal on Imaging Sciences*, 3(4):1048–1074, Dec 2010. 1, 2

[17] S. Nowozin and C. H. Lampert. Global connectivity potentials for random field models. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 818–825, Jun 2009. 1, 2, 3, 5

[18] M. Rempfler, B. Andres, and B. Menze. The minimum cost connected subgraph problem in medical image analysis. In *Medical Image Computing and Computer Assisted Intervention - MICCAI (3)*, pages 397–405, Oct 2016. 2, 4, 5

[19] D. Stutz, A. Hermans, and B. Leibe. Superpixels: An evaluation of the state-of-the-art. *Computer Vision and Image Understanding*, 166(Supplement C), Jan 2018. 5

[20] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):1286–80, Jun 2008. 1

[21] S. Vicente, V. Kolmogorov, and C. Rother. Graph cut based image segmentation with connectivity priors. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, Jun 2008. 1, 2

[22] J. Yedidia, W. Freeman, and Y. Weiss. Understanding belief propagation and its generalizations. In *Exploring artificial intelligence in the new millennium*, pages 239–269. Morgan Kaufmann Publishers Inc., 2003. 2