Partial Optimality via Iterative Pruning for the Potts Model

Paul Swoboda[1], Bogdan Savchynskyy[2], Jörg Kappes[1] and Christoph Schnörr[1,2]

[1]Image and Pattern Analysis Group, [2]Heidelberg Collaboratory for Image Processing
University of Heidelberg, Germany

Abstract. We propose a novel method to obtain a part of an optimal *non-relaxed integral* solution for energy minimization problems with Potts interactions, known also as the minimal partition problem. The method empirically outperforms previous approaches like MQPBO and Kovtun's method in most of our test instances and especially in hard ones. As a starting point our approach uses the solution of a commonly accepted convex relaxation of the problem. This solution is then iteratively pruned until our criterion for partial optimality is satisfied. Due to its generality our method can employ any solver for the considered relaxed problem.

1 Introduction

1.1 Problem Formulation

Continuous model. Consider the minimal partition problem

$$\min_{u \in BV(\Omega;\{1,\ldots,k\})} \int_\Omega |Du| + W(x,u(x))dx \; . \tag{1.1}$$

This problem and approximation algorithms for it are discussed in [12] for the discrete case, and in [7] and [15] for the continuous case.

Minimizing discretizations of the above problem is NP-hard for $n \geq 3$, therefore it is common to resort to a convex relaxation. Introduce

$$\begin{aligned} u^i(x) \geq 0 \; , \; i = 1 \ldots, k \, , \\ \textstyle\sum_{i=1}^k u^i(x) = 1 \; , \; x \in \Omega \, , \end{aligned} \tag{1.2}$$

or equivalently $u(x) \in \Delta_k$, $x \in \Omega$, and minimize (1.1) over (1.2). In general a minimizer u^* of the relaxed problem will not be binary anymore, but for some $x \in \Omega$ it may still hold true. A natural question is: Is there a minimizer $\tilde{u}$ of the original NP-hard problem (1.1) and such a subset $A \subset \Omega$, that $\tilde{u}(x) = u^*(x)$ for $x \in A$? In other words, is u^* partially optimal or *persistent* on some set A? How can we determine such a set A?

Finding persistency is not only theoretically interesting, but it also allows in many cases to solve the problem w.r.t. the remaining non-persistent variables with other methods as done in [3] and thereby to obtain its complete globally optimal solution. Moreover, solving the problem with respect to the non-persistent variables is simplified by the fact, that the latter are often weakly connected or/and form small connected components.

Discrete model. For solving problem (1.1) in practice, one must discretize it. There are many possible ways to do so, see e.g. [7] or [15]. We consider a discretization, which introduces anisotropies, but can be stated for general graphs $G = (\Omega, E)$:

$$J(u) = \sum_{a \in \Omega} \langle c(a), u(a) \rangle + \sum_{(a,b) \in E} \sum_{l=1}^{k} \alpha_{a,b} |u^l(a) - u^l(b)| \qquad (1.3)$$

with $\alpha_{a,b} > 0$, $(a,b) \in E$, $c(a) \in \mathbb{R}^k$ and $u(a) \in \{0,1\}^k$, $a \in \Omega$, satisfies additionally (1.2). The discrete problem (1.3) is also known as a Potts model.

Note that for a grid graph, uniform weights $\alpha_{a,b}$ and c being a local average of W, this is a particular discretization of the minimal partition problem (1.1). One can systematically approximate the minimal partition problem with graphs of higher connectivity with the method presented in [6], while still solving a problem of the form (1.3).

1.2   Related Work and Contribution

The task of finding persistent variables in labeling problems has been studied and many approaches have been proposed [5, 9, 10, 13, 14, 16, 17, 20]. To our best knowledge the earliest paper concerning itself with persistency is [16], which states a persistency criterion for the stable set problem and verifies it for every solution of a certain relaxation, which the roof duality method in [5] uses and which is also the basis for the well known QPBO-algorithm [5, 17]. Roof Duality has been extended for Multi-Label problems in [13, 20] and for higher order binary problems in [10]. A different approach, specialized for Potts models, is pursued in [14], where possible labelings are tested for persistency.

MQPBO. In [13] the authors transform the multilabeling problem into a quadratic binary problem. Their transformation is dependent upon choosing a label order and their results are so as well. It is not known how to choose an optimal label ordering to obtain the maximum number of persistent variables. For actually solving their problem they use a relaxation which is an outer relaxation of the local polytope [18, Prop. 1]. One can show that *the relaxation we use is strictly tighter than theirs and our approach also generalizes to tighter relaxations as well*. Experimentally we are able to label a much higher percentage of points persistently. In case of high regularization weights our approach can *determine a substantially higher number of persistently labeled variables*. While the model [18] can solve more general interaction potentials, it needs significantly more memory for the interaction terms. For the Potts model our approach consumes *substantially less memory*, if a suitable algorithm for solving the relaxation is used.

Kovtun's Approach [14] consists in searching for partially optimal labelings by constructing auxiliary problems, solving these and testing for persistent variables. Each of the auxiliary problems is however less tight than the relaxation we use. Also experimentally, *we could usually label more variables than this method*.

1.3   Organization

We present

- a new persistency criterion for ensuring a labeling to be partially optimal, see Section 2,

- an algorithm for finding the provably biggest labeling, such that the persistency criterion is satisfied, see Section 2,
- experimental validation of our approach and a comparison to existing methods and new ways to use them together, see Section 3.

For the sake of clarity of presentation we have moved the proofs of all theoretical statements to Section 5.

1.4   Notation

We reserve the variable k for the number of classes in problem (1.3) and denote by $e_i = (0,\dots,0,1,0,\dots,0)^\top$ the ith basis vector of $\mathbb{R}^k$ with a 1 in its ith component. Let $\mathscr{E}_k := \{e_1,\dots,e_k\}$ contain the k unit basis vectors and denote by $\Delta_k = \mathrm{conv}(\mathscr{E}_k)$ its convex hull. For notational convenience we introduce two labeling spaces. Let

$$D = \{u : \Omega \to \mathscr{E}_k\}, \tag{1.4a}$$

$$D' = \{u : \Omega \to \Delta_k\}. \tag{1.4b}$$

The set D consists of all integral labelings and is nonconvex. The set D' is its convex hull, consists of functions defined by (1.2) and is used as feasible set in the relaxations.

The characteristic function of a set C is defined as $\delta_C(x) = \begin{cases} 0 & ,x \in C \\ \infty & ,x \notin C \end{cases}$.

For a subset of nodes $A \subset \Omega$ let the functional restricted to A be

$$J_A(u) = \sum_{a \in A}\langle u(a),c(a)\rangle + \sum_{(a,b) \in E}\sum_{l=1}^k \alpha_{a,b}|u^l(a) - u^l(b)| \ . \tag{1.5}$$

For $A \subset \Omega$ let its boundary be given by

$$\partial A = \{a \in A \ : \ \exists b \in \Omega \setminus A \text{ s.t. } (a,b) \in E\} \ . \tag{1.6}$$

Definition 1. *For a boundary term $w : \partial A \to \mathscr{E}_k$ let the functional restricted to A with the boundary term w be defined by*

$$J_{A,w}(u) = J_A(u) + \sum_{(a,b) \in E : a \in A, b \in \Omega \setminus A} 2\alpha_{a,b}\langle w(a),u(a)\rangle \ . \tag{1.7}$$

2   Persistency for the Discretized Potts Problem

First we propose a criterion for partial optimality. Suppose we have found an integer labeling on a set A, optimal for J_A, which is not affected by what happens on its complement $\Omega \setminus A$. Then it is immediate that the labeling is partially optimal. We propose in the following Lemma 1 a sufficient condition for this situation. More specifically, a binary minimizer of $J_{A,w} + \delta_{D'}$, which conforms to the boundary conditions w on ∂A is partially optimal.

Lemma 1. *Let $w : \partial A \to \mathscr{E}_k$ be given. Suppose $u^* : A \to \mathscr{E}_k$ is optimal for the functional*

$$J_{A,w}(u) + \delta_D(u) \tag{2.1}$$

and $u^*(a) = w(a) \ \forall a \in \partial A$. Then there exists a labeling $\tilde{u} : \Omega \to \mathscr{E}_k$ which is optimal for $J(u) + \delta_D$ and such that $\tilde{u}_{|A} = u^*$.
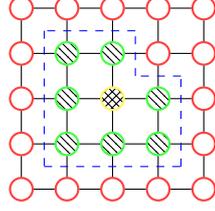
Fig. 1. An exemplaric graph for the setting of Lemma 1. The blue dashed line encloses the set A, the green nodes with the diagonal pattern have a boundary labeling w while the yellow node with the crosshatch pattern is in the interior of A and thus has no boundary labeling.

For computational purposes we must relax the functional $J_{A,w} + \delta_D$. Still the statement of Lemma 1 essentially holds:

Corollary 1. *Let $w : \partial A \to \mathscr{E}_k$ be given. Suppose the integral labeling $u^* : A \to \mathscr{E}_k$, is optimal for the relaxed functional*

$$J_{A,w}(u) + \delta_{D'}, \qquad (2.2)$$

and the boundary condition

$$u^*(a) = w(a),\ a \in \partial A, \qquad (2.3)$$

holds. Then there exists $\tilde{u} : \Omega \to \mathscr{E}_k$ optimal for $J(u) + \delta_D$ such that $\tilde{u}_{|A} = u^*$, i.e. u^* is partially optimal.

Corollary 1 forms the basis for Algorithm 1, which constructs a set of persistent variables. The algorithm is initialized with the whole set Ω and recursively shrinks it by removing variables taking non-integral values or not conforming to the boundary condition (2.3). The process stops, when there is an optimizer u^* fulfilling the conditions of Corollary 1 for the remaining set A^*.

Algorithm 1: Finding persistent variables

Data: $G = (\Omega, E)$, $c : \Omega \to \mathbb{R}^k$,
$\qquad \alpha_{a,b} \in \mathbb{R}_+ : (a,b) \in E$
Result: $A^* \subset \Omega$, $u^* : A^* \to \mathscr{E}_k$
Initialize:
$A^0 = \Omega$;
$\tilde{w}^0 = 0$;
Choose a $\tilde{u}^0 \in \operatorname{argmin}_u J_{A^0, \tilde{w}^0}(u) + \delta_{D'}$;
$t = 1$;
while $\tilde{u}^t \notin D$ or $\tilde{u}^t_{|\partial A^t} \neq \tilde{w}^t$ **do**
$\qquad W^t = \{b \in \partial A^{t-1} : \tilde{u}^{t-1}(b) \neq \tilde{w}^{t-1}(b)\}$;
$\qquad A^t = \{a \in A^{t-1} : \tilde{u}^{t-1}(a) \in \mathscr{E}_k\} \backslash W^t$;
$\qquad \tilde{w}^t = \tilde{u}^t_{|\partial A^t}$;
$\qquad$ Choose a $\tilde{u}^t \in \operatorname{argmin}_u J_{A^t, \tilde{w}^t}(u) + \delta_{D'}$;
$\qquad t = t + 1$;
end
$A^* = A^t$;
$u^* = \tilde{u}^t$;

Solve the relaxed problem over Ω without boundary conditions

Shrink the set A^t by removing variables taking non-integral values or not conforming to the current boundary condition

---

In each iteration the set A^t shrinks. Since Ω is finite, the algorithm converges in at most $|\Omega|$ steps. If the algorithm stops, then we have that

$$u^* \in \operatorname{argmin}_u J_{A^*,w} + \delta_{D'}, \qquad (2.4)$$

$w = u^*_{|\partial A^*}$ and $u^* \in D$. Hence u^*, w and A fulfill the conditions of Corollary 1, which proves persistency. In what follows we will show, that under a mild technical assumption Algorithm 1 is in some sense optimal, i.e. it delivers the *greatest* persistent set conforming to Corollary 1.

Assumption 1 *There is a unique solution of $J_{A^t,w^t} + \delta_{D'}$ for each A^t and each $w^t : \partial A^t \to \mathscr{E}_k$ obtained during iterations of Algorithm 1.*

Definition 2. *A subset $A \subset \Omega$ is the* greatest *persistent set for the functional $J + \delta_{D'}$, if for all other sets $A' \subset \Omega$ fulfilling the conditions of Corollary 1, it follows that $A' \subset A$.*

Theorem 1. *Under Assumption 1 Algorithm 1 returns the greatest persistent set A^*.*

Remark 1. If Assumption 1 does not hold, then Algorithm 1 is not deterministic and the obtained set A^* is not necessarily the greatest persistent set. The simplest example of such a situation occurs if the relaxation $J + \delta_{D'}$ 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 these entries of the solution will not be marked as persistent.

Remark 2. Algorithms similar to Algorithm 1 can be applied also to tighter relaxations, e.g. when one includes cycle inequalities similar to [19]. All our results are independent of the specific relaxation and the method one uses to optimize the relaxed problems (2.2). One can show that the persistent variables one obtains with a tighter relaxation are a superset of persistent variables one gets from a weaker relaxation.

At first glance it may seem that Algorithm 1 is not very efficient, due to the need to compute optimal solutions to possibly many slightly differing problems in the iterations of the while loop in Algorithm 1. The procedure is however significantly accelerated with a warm start, i.e. initializing the algorithm with the variables from the previous iteration. We discuss time issues in Section 3.

3   Experiments

We compare to the following methods:

– **MQPBO:** see section 1.2 and [13, 18]. As in [13] we fix a label order before running the MQPBO algorithm. The labels are ordered according to the strength of the local data term. Note that contrary to our and Kovtun's approach, MQPBO can also detect persistency for labels which will *not* belong to any optimal solution.
– **Kovtun:** see section 1.2 and [14].
– **KMQPBO:** We first run Kovtun's method and then we run MQPBO on the variables which could not be found persistent by Kovtun's method.

– **KMQPBO100:** We first run Kovtun's method and then run sequentially 100 iteration of the MQPBO algorithm with a randomly sampled label order and accumulate persistent variables.

Note that KMQPBO100 is an improvement upon the remaining (K)MQPBO methods and obviously also over Kovtun's method. MQPBO's results really depend upon the label ordering and therefore finding a good or somehow optimal label order for specific problems remains an interesting task. This can be seen in the experiments, where KMQPBO100 outperforms KMQPBO. While it would be favourable for KMQPBO to optimize over all possible permutation per variable, this is computationally not tractable. The problem of choosing the best label order has not been dealt with in literature, as far as we know.

Our method uses the fast primal-dual method from [8] for minimizing the relaxations $J_{A^t,w^t} + \delta_{D'}$.

For illustrating the strength of our method we present three datasets for the Potts model. The datasets are explained in greater detail in [11] and are available on the accompanying website [2]. The first dataset contains segmentation problems, for which we are given a few prototypical color vectors. The distance between each pixel's color value and each prototypical vector is measured, thereby obtaining the local data term $c(a)$ in (1.3). The regularization strength $\alpha_{a,b}$ in (1.3) was set uniformly. See Table 1 for the results. The number of classes is written in parentheses behind the instance name and the numbers denote the percentage of persistently labelled variables. The image dimensions are usually 360×240 or slightly less. We have also included a time plot for the clownfish dataset in Table 1, see Fig. 2.

Dataset	Our method	KMQPBO	KMQPBO100	Kovtun	MQPBO
clownfish (12)	**0.9852**	0.7659	0.9495	0.7411	0.0467
crops (12)	**0.9308**	0.6486	0.8803	0.6470	0.0071
fourcolors(4)	**0.9993**	0.6952	0.7010	0.6952	0.0
lake (12)	**0.9998**	0.7613	0.9362	0.7487	0.0665
palm (12)	**0.8514**	0.6866	0.7192	0.6865	0.0
penguin (8)	**0.9999**	0.9240	0.9471	0.9199	0.0103
peacock (12)	0.1035	0.0559	**0.1234**	0.0559	0.0
snail (3)	**0.9997**	0.9786	0.9819	0.9778	0.5835
strawberry-glass (12)	**0.9639**	0.5502	0.5997	0.5499	0.0

Table 1. Results for segmentation problems with prototypical color vectors. Entries in the table denote the percentage of persistently labelled variables. The numbers in parentheses behind the dataset name denote the number k of classes.

The second dataset consists of segmentation problems of a simulated brain scan with 5 prototypical vectors. The brain images were generated with the simulator [1]. The local data terms $c(a)$ in (1.3) were computed as for the first dataset and the regularization strength $\alpha_{a,b}$ in (1.3) was set uniformly as well. Instances can become very huge due to the volumes being three-dimensional. See Table 2 for results. The dimensions of the
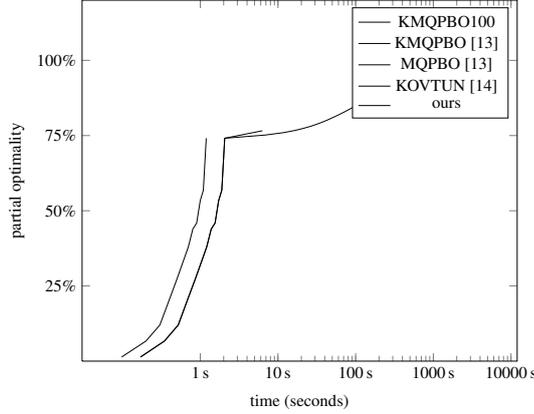
Fig. 2. Percentage of partial optimal variables of the 5 compared methods over time for the clownfish dataset. The method of Kovtun provides partial optimality very fast. KMQPBO100 finds more and more partially optimal variables, which illustrates that the performance of MQBPO depends on the label order. While our pruning method gives the best result, our current research implementation is not competitive with respect to time.

problems are denoted in the left column, while the entries denote the percentage of the variables determined to be persistent.

Dataset	Our method	KMQPBO	KMQPBO100	Kovtun	MQPBO
$181 \times 217 \times 20$	0.9968	0.9993	**0.9994**	0.9235	0.3886
$181 \times 217 \times 26$	0.9969	**1**	0.9996	0.9322	0.3992
$181 \times 217 \times 36$	**0.9967**	†	†	0.9363	0.4020
$181 \times 217 \times 60$	**0.9952**	†	†	0.9496	0.4106

Table 2. Results for the simulated brain scan dataset. Entries in the table denote the percentage of persistently labelled variables. The numbers in the left column denote the image dimensions. The number k of classes is 5. Entries denoted by † indicate that the instance could not be solved with the specified method for implementation reasons.

The third dataset consists of object segmentation problems with the local data terms $c(a)$ in (1.3) denoting the probability of pixels to belong to object classes. The regularization strength $\alpha_{a,b}$ in (1.3) is chosen to be inversely proportional to the image gradient. The data was taken from [4]. It turned out that the relaxation we use was already tight for the data. Hence we could determine all variables in the initial run. This illustrates experimentally, that the relaxations used in Kovtun's method and in MQPBO are less tight than our relaxation. See Table 3 for results.

Most often we could label over 95% of all variables persistently with our method and outperform the other tested approaches. Note that *we can use an arbitrary algorithm for solving the problem* (2.2) *in contrast to the approaches based on roof duality*. It is very

8   Paul Swoboda[1], Bogdan Savchynskyy[2], Jörg Kappes[1] and Christoph Schnörr[1,2]

Dataset	Our method	KMQPBO	KMQPBO100	Kovtun	MQPBO
Plane (4)	**1**	0.9833	0.9833	0.9833	0.0002
Bikes (5)	**1**	0.9570	0.9570	0.9569	0.0
Road (5)	**1**	0.9579	0.9579	0.9579	0.0
Building (7)	**1**	0.8051	0.8053	0.8051	0.0
Car (8)	**1**	0.9902	0.9904	0.9902	0.0002

Table 3. Results for the object segmentation dataset. Entries in the table denote the percentage of persistently labelled variables. The numbers in the left column the number of classes k.

noteworthy, that KMQPBO and KMQPBO100 outperform our approach in the brain scan dataset as well as in one instance of the color segmentation dataset. Although the relaxation we use is tighter than MQPBO's, all the integral variables MQPBO obtains are persistent. In contrast, only a subset of the integral variables of the solution to the relaxation we use are found persistent. Hence, it may occur that in some instances (K)MQPBO(100) can label more variables persistently although our approach yields more integral variables which however cannot be proved persistent. The same reasoning applies to Kovtun's method, so possibly some variables could be found persistent in Kovtun's method and hence also in KMQPBO and KMQPBO100 which could not be verified to be persistent in our approach.

Kovtun's method from [14] is very fast, usually faster than solving the relaxation of problem (1.3). Therefore using a layered approach by first applying Kovtun's Partial Optimal Labeling Search from [14] and then applying our approach on the remaining variables will result in at least the same number of persistent variables while still retaining a very fast runtime. For properly comparing the approaches however, we have used our method on its own.

4   Conclusion and Outlook

We have presented a method for finding persistent variables for the Potts model, which outperforms other approaches to this problem with respect to the number of persistent variables found. The presented method can use an arbitrary algorithm for minimizing a relaxed labeling problem and generalizes to tighter relaxations as well.

In future we will address the problem of finding persistent variables for arbitrary graphical models and the discretized minimal partition problem with better discretizations.

References

1. Brainweb: Simulated brain database. http://brainweb.bic.mni.mcgill.ca/brainweb/.

2. OpenGM inference library. http://hci.iwr.uni-heidelberg.de/opengm2/.
3. K. Alahari, P. Kohli, and P. H. S. Torr. Reduce, reuse & recycle: Efficiently solving multi-label MRFs. In *CVPR*, 2008.
4. K. Alahari, P. Kohli, and P. H. S. Torr. Dynamic hybrid algorithms for MAP inference in discrete MRFs. *PAMI*, 32(10):1846–1857, 2010.
5. E. Boros and P. L. Hammer. Pseudo-Boolean optimization. *Discrete Applied Mathematics*, 123(1–3):155–225, 2002.
6. Yuri Boykov and Vladimir Kolmogorov. Computing geodesics and minimal surfaces via graph cuts. In *ICCV*, pages 26–33. IEEE Computer Society, 2003.
7. A. Chambolle, D. Cremers, and T. Pock. A convex approach for computing minimal partitions. Technical report, Centre des Mathematiques Appliquees, Ecole Polytechnique, Palaiseau, Paris, France, 2008.
8. A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.
9. P.L. Hammer, P. Hansen, and B. Simeone. Roof duality, complementation and persistency in quadratic 0-1 optimization. *Math. Programming*, 28:121–155, 1984.
10. Fredrik Kahl and Petter Strandmark. Generalized roof duality. *Discrete Applied Mathematics*, 160(16-17):2419–2434, 2012.
11. 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.
12. J. Kleinberg and É. Tardos. Approximation algorithms for classification problems with pairwise relationships: metric labeling and Markov random fields. *J. ACM*, 49(5):616–639, September 2002.
13. P. Kohli, A. Shekhovtsov, C. Rother, V. Kolmogorov, and P. Torr. On partial optimality in multi-label MRFs. In *ICML*, pages 480–487, 2008.
14. Ivan Kovtun. Partial optimal labeling search for a NP-hard subclass of (max, +) problems. In *DAGM*, pages 402–409, 2003.
15. J. Lellmann and C. Schnörr. Continuous multiclass labeling approaches and algorithms. *SIAM J. Imag. Sci.*, 4(4):1049–1096, 2011.
16. G. L. Nemhauser and L. E. Trotter. Vertex packings: Structural properties and algorithms. *Mathematical Programming*, 8:232–248, 1975. 10.1007/BF01580444.
17. Carsten Rother, Vladimir Kolmogorov, Victor S. Lempitsky, and Martin Szummer. Optimizing binary MRFs via extended roof duality. In *CVPR*, 2007.
18. A. Shekhovtsov, V. Kolmogorov, P. Kohli, V. Hlavac, C. Rother, and P. Torr. LP-relaxation of binarized energy minimization. Research Report CTU–CMP–2007–27, Czech Technical University, 2008.
19. D. Sontag. *Approximate Inference in Graphical Models using LP Relaxations*. PhD thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, 2010.
20. T. Windheuser, H. Ishikawa, and D. Cremers. Generalized Roof Duality for Multi-Label Optimization: Optimal Lower Bounds and Persistency. In *ECCV*, Firenze, Italy, Oct. 2012.

5 Appendix

Lemma 1.

Proof. Let $\bar{u} : \Omega \backslash A \to \mathscr{E}_k$ be optimal for the functional

$$J_{\Omega \backslash A}(u) - \sum_{a \in (\Omega \backslash A), b \in A, (a,b) \in E} 2\alpha_{a,b} \langle w(b), u(a) \rangle + \delta_D. \tag{5.1}$$

Then

$$\tilde{u}(a) = \begin{cases} u^*(a) & ,a \in A \\ \overline{u}(a) & ,a \notin A \end{cases} \tag{5.2}$$

is optimal for $J + \delta_D$. Let $u' : \Omega \to \mathscr{E}_k$ be another labeling. If we will show, that $J(\tilde{u}) \le J(u')$, it will prove the lemma. Indeed, taking into account that $u^*(a) = w(a), a \in \partial A$ we have

$$
\begin{aligned}
&J(\tilde{u}) \\
&= J_A(u^0) + J_{\Omega \setminus A}(\overline{u}) + \sum_{b \in \Omega \setminus A, a \in A, (a,b) \in E} \sum_{l=1}^k \alpha_{a,b} |\tilde{u}^l(a) - \tilde{u}^l(b)| \\
&= J_A(u^0) + J_{\Omega \setminus A}(\overline{u}) + \sum_{b \in \Omega \setminus A, a \in A, (a,b) \in E} \sum_{l=1}^k \alpha_{a,b} |w^l(a) - \tilde{u}^l(b)| \\
&= J_A(u^0) + J_{\Omega \setminus A}(\overline{u}) + \sum_{b \in \Omega \setminus A, a \in A, (a,b) \in E} 2\alpha_{a,b}(1 - \mathbb{1}_{\overline{u}(b)=w(a)}) \\
&= J_A(u^0) + J_{\Omega \setminus A}(\overline{u}) + \sum_{b \in \Omega \setminus A, a \in A, (a,b) \in E} 2\alpha_{a,b} \langle w(a), u^0(a) - \overline{u}(b) \rangle \\
&\le J_A(u'_{|A}) + J_{\Omega \setminus A}(u'_{|\Omega \setminus A}) + \sum_{b \in \Omega \setminus A, a \in A, (a,b) \in E} 2\alpha_{a,b} \langle w(a), u'(a) - u'(b) \rangle \\
&= J_A(u'_{|A}) + J_{\Omega \setminus A}(u'_{|\Omega \setminus A}) + \sum_{b \in \Omega \setminus A, a \in A, (a,b) \in E} 2\alpha_{a,b}(\mathbb{1}_{w(a)=u'(a)} - \mathbb{1}_{w(a)=u'(b)}) \\
&\le J_A(u'_{|A}) + J_{\Omega \setminus A}(u'_{|\Omega \setminus A}) + \sum_{b \in \Omega \setminus A, a \in A, (a,b) \in E} \alpha_{a,b} \sum_{l=1}^k |u'(a) - u'(b)| \\
&= J(u'),
\end{aligned} \tag{5.3}
$$

which finalizes the proof.

Corollary 1.

Proof. $u^* \in \operatorname{argmin} J_{A,w} + \delta_{D'}$ and $u^* \in D \Rightarrow u^* \in \operatorname{argmin} J_{A,w} + \delta_D$. Now apply Lemma 1 and get statement of the corollary.

To prove Theorem 1 we will require the following three lemmas.

Lemma 2. *Let $\alpha \in \mathscr{E}_k$, $\beta \in \Delta_k$. Then*

$$2\langle \alpha, \alpha - \beta \rangle = \sum_{l=1}^k |\alpha^l - \beta^l| \tag{5.4}$$

Proof. Without loss of generality assume $\alpha = e_i$. Then

$$
\begin{aligned}
2\langle \alpha, \alpha - \beta \rangle &= 2 - 2\beta^i = 1 - \beta^i + 1 - \beta^i \\
&= 1 - \beta^i + \sum_{l \ne i} b^l = |1 - \beta^i| + \sum_{l \ne i} |b^l| = \sum_{l=1}^k |\alpha^l - \beta^l|
\end{aligned} \tag{5.5}
$$

Lemma 3. *Let $\alpha, \beta \in \Delta_k$ and $e_i \in \mathscr{E}_k$. Then*

$$2\langle e_i, \alpha - \beta \rangle \le \sum_{l=1}^k |\alpha^l - \beta^l|. \tag{5.6}$$

Proof.

$$
\begin{aligned}
2\langle e_i, \alpha - \beta \rangle &= 2(\alpha^i - \beta^i) = (\alpha^i - \beta^i) + (1 - \sum_{l \ne i} a^l) - (1 - \sum_{l \ne i} \beta^l) \\
&= (\alpha^i - \beta^i) + \sum_{l \ne i} (\beta^l - a^l) \le \sum_{l=1}^k |\alpha^l - \beta^l|.
\end{aligned} \tag{5.7}
$$

Note that equality holds in (5.6) iff $a^i \ge \beta^i$ and $\alpha^j \le \beta^j$ for all $j \ne i$.

Lemma 4. *Let $A \subset \Omega$, $\bar{u}: A \to \mathscr{E}_k$ and $\overline{w}: \partial A \to \mathscr{E}_k$ satisfy the persistency criterion of Corollary 1.*

Then for any $B \supset A$ and boundary condition $\tilde{w}$ satisfying $\tilde{w}(a) = \overline{w}(a)$ for all $a \in \partial A \cap \partial B$, such $\tilde{u}$ exists, that

$$\tilde{u} \in \operatorname{argmin}_u J_{B,\tilde{w}}(u) + \delta_{D'} \tag{5.8}$$

and $\tilde{u}_{|A} = \bar{u}$ and $\tilde{w}(a) = \tilde{u}(a)$ for $a \in \partial A$.

Proof. Let u' be optimal for the functional

$$J_{B \setminus A}(u) - \sum_{(a,b) \in E : \, b \in B \setminus A, a \in A} 2\alpha_{a,b} \langle w(a), u(b) \rangle$$

$$+ \sum_{(a,b) \in E : \, a \in B \setminus A, b \in B \setminus \Omega} 2\alpha_{a,b} \langle \tilde{w}(a), u(a) \rangle + \delta_{D'}. \tag{5.9}$$

Let

$$\tilde{u}(a) = \begin{cases} \bar{u}(a) & , a \in A \\ u'(a) & , a \in B \setminus A. \end{cases} \tag{5.10}$$

The lemma will be proved when we show that $\tilde{u} \in \operatorname{argmin}_u J_{B,\tilde{w}}(u) + \delta_{D'}(u)$. For this let $\hat{u}$ be arbitrary relaxed labeling from $\in \delta_{D'}$. The following inequalities then hold:

$$J_{B,\tilde{w}}(\tilde{u}) \tag{5.11}$$

$$= J_A(\bar{u}) + J_{B \setminus A}(u') \tag{5.12}$$

$$+ \sum_{(a,b) \in E : \, a \in A, b \in B \setminus A} \sum_{l=1}^{k} \alpha_{a,b} |\bar{u}^l(a) - (u')^l(b)| \tag{5.13}$$

$$+ \sum_{(a,b) \in E : \, a \in B, b \in \Omega \setminus B} 2\alpha_{a,b} \langle \tilde{w}(a), \tilde{u}(a) \rangle \tag{5.14}$$

$$= J_A(\bar{u}) + J_{B \setminus A}(u') \tag{5.15}$$

$$+ \sum_{(a,b) \in E : \, a \in A, b \in B \setminus A} \sum_{l=1}^{k} \alpha_{a,b} |\bar{u}^l(a) - (u')^l(b)| \tag{5.16}$$

$$+ \sum_{(a,b) \in E : \, a \in A, b \in \Omega \setminus B} 2\alpha_{a,b} \langle \overline{w}(a), \bar{u}(a) \rangle \tag{5.17}$$

$$+ \sum_{(a,b) \in E : \, a \in B \setminus A, b \in \Omega \setminus B} 2\alpha_{a,b} \langle \tilde{w}(a), u'(a) \rangle \tag{5.18}$$

$$\overset{(*)}{=} J_A(\bar{u}) + J_{B \setminus A}(u') \tag{5.19}$$

$$+ \sum_{(a,b) \in E : \, a \in A, b \in B \setminus A} 2\alpha_{a,b} \langle \overline{w}(a), \bar{u}(a) - u'(b) \rangle \tag{5.20}$$

$$+ \sum_{(a,b) \in E : \, a \in A, b \in \Omega \setminus B} 2\alpha_{a,b} \langle \overline{w}(a), \bar{u}(a) \rangle \tag{5.21}$$

$$+ \sum_{(a,b) \in E : \, a \in B \setminus A, b \in \Omega \setminus B} 2\alpha_{a,b} \langle \tilde{w}(a), u'(a) \rangle \tag{5.22}$$

$$= J_A(\bar{u}) + J_{B \setminus A}(u') \tag{5.23}$$

$$+ \sum_{(a,b) \in E : \, a \in A, b \in \Omega \setminus A} 2\alpha_{a,b} \langle \overline{w}(a), \bar{u}(a) \rangle \tag{5.24}$$

$$- \sum_{(a,b) \in E : \, a \in A, b \in B \setminus A} 2\alpha_{a,b} \langle \overline{w}(a), u'(b) \rangle \tag{5.25}$$

$$+ \sum_{(a,b) \in E : \, a \in B \setminus A, b \in \Omega \setminus B} 2\alpha_{a,b} \langle \tilde{w}(a), u'(a) \rangle \tag{5.26}$$

$$\overset{(**)}{\leq} J_A(\hat{u}_{|A}) + J_{B \setminus A}(\hat{u}_{|B \setminus A}) \tag{5.27}$$

$$+ \sum_{(a,b) \in E : \, a \in A, b \in \Omega \setminus A} 2\alpha_{a,b} \langle \overline{w}(a), \hat{u}(a) \rangle \tag{5.28}$$

$$- \sum_{(a,b) \in E : \, a \in A, b \in B \setminus A} 2\alpha_{a,b} \langle \overline{w}(a), \hat{u}(b) \rangle \tag{5.29}$$

$$+ \sum_{(a,b) \in E : \, a \in B \setminus A, b \in \Omega \setminus B} 2\alpha_{a,b} \langle \tilde{w}(a), \hat{u}(a) \rangle \tag{5.30}$$

$$= J_A(\tilde{u}_{|A}) + J_{B \setminus A}(\hat{u}_{|B \setminus A}) \tag{5.31}$$

$$+ \sum_{(a,b) \in E : \, a \in A, b \in B \setminus A} 2\alpha_{a,b} \langle \overline{w}(a), \hat{u}(a) - \hat{u}(b) \rangle \tag{5.32}$$

$$+ \sum_{(a,b) \in E : \, a \in A, b \in \Omega \setminus B} 2\alpha_{a,b} \langle \overline{w}(a), \hat{u}(a) \rangle \tag{5.33}$$

$$+ \sum_{(a,b) \in E : \, a \in B \setminus A, b \in \Omega \setminus B} 2\alpha_{a,b} \langle \tilde{w}(a), \hat{u}(a) \rangle \tag{5.34}$$

$$\overset{(***)}{\leq} J_A(\hat{u}_{|A}) + J_{B \setminus A}(\hat{u}_{|B \setminus A}) \tag{5.35}$$

$$+ \sum_{(a,b) \in E : \, a \in A, b \in B \setminus A} \sum_{l=1}^{k} \alpha_{a,b} |\hat{u}^l(a) - \hat{u}^l(b)| \tag{5.36}$$

$$+ \sum_{(a,b) \in E : \, a \in B, b \in \Omega \setminus B} 2\alpha_{a,b} \langle \tilde{w}(a), \hat{u}(a) \rangle \tag{5.37}$$

$$= J_{B, \tilde{w}}(\hat{u}) \,, \tag{5.38}$$

where $(*)$ is due to lemma 2, $(**)$ is due to the optimality of $\overline{u}$ and u' for the respective functionals and $(***)$ is due to lemma 3.

Theorem 1

Proof. We prove the statement that A^* is the greatest persistent set by showing the following claim to hold true:

Claim: Assume that for some $\overline{A} \subset \Omega$ there exists a persistent labeling $\overline{u}$. Then in each iteration of the algorithm $\overline{A} \subset A^i$ holds. Furthermore $\overline{u} = u^*_{|\overline{A}}$ and hence $\overline{w}(a) = w^*(a)$ for $a \in \partial \overline{A} \cap \partial A^*$, where $w^* = u^*_{|\partial A^*}$.

In the initialization step the claim clearly holds true, since $A^0 = \Omega$ and Lemma 4 ensures that there exists $\tilde{u}^0 \in \text{argmin} J_\Omega + \delta_{D'}$ such that $\overline{u} = \tilde{u}^0_{|\overline{A}}$ and $\overline{w}(a) = \tilde{w}^0(a)$ for all $a \in \partial \overline{A} \cap \partial \Omega = \varnothing$ is an empty condition. Finally assumption 1 gives us that there is only one such minimizer $\tilde{u}^0$, so the claim holds initially.

Now assume the claim to hold for $i - 1$. We need to show that it also holds for i. For this just invoke Lemma 4 with $A = \overline{A}$, $B = A^{i-1}$ and $\tilde{w} = \tilde{w}^{i-1}$. The conditions of Lemma 4 hold by assumption on $i - 1$. Lemma 4 now ensures existence of $\tilde{u}^i \in \text{argmin}_u J_{A^{i-1}, \tilde{w}^{i-1}}(u) + \delta_{D'}(u)$ with the required properties. Again by Assumption 1, $\tilde{u}^i$ is unique, so we are done.

Inspecting the proof of the claim above, we see that Assumption 1 is necessary because otherwise the labels for nodes in $\overline{A}$ could possibly change during iterations of the algorithm or be convex sums of optimal persistent labellings in which case they would be discarded from the sets A^i at some point.