Multi-label cooperative cuts

Stefanie Jegelka
Max Planck Institute for Intelligent Systems
jegelka@tuebingen.mpg.de

Jeff Bilmes
University of Washington
bilmes@ee.washington.edu

Abstract

Recently, a family of global, non-submodular energy functions has been proposed that is expressed as coupling edges in a graph cut. This formulation provides a rich modelling framework and also leads to efficient approximate inference algorithms. So far, the results addressed binary random variables. Here, we extend these results to the multi-label case, and combine edge coupling with move-making algorithms.

1. Introduction

Most energy functions and probabilistic models in computer vision have relied on properties such as submodularity, sparsity or low-order potentials to keep the energy minimization or MAP (maximum a posteriori) inference problem tractable. In particular, submodular pairwise potentials lead to the successful graph cut methods [1, 9] that are efficient and exact. Indeed, as a result, higher-order potentials are often eventually expressed as pairwise submodular energy functions with additional variables, and optimized using graph cuts. Despite their flexibility, however, the expressiveness of such graph-representable energies is limited if one restricts the number of additional nodes and edges. Thus, recent research has aimed to identify practically manageable higher-order potentials [3, 7, 6, 10] or non-submodular energies [8], and efficient optimization methods for those.

In [5], we define a family of non-submodular, global energies that makes use of a different type of structure. It still uses a graph structure, and exploits submodularity indirectly. The crucial idea is that graph cut based energy functions can be significantly enhanced if the cost of a cut is not the sum of the edge weights, but a richer function that allows interactions between edges. In short, edges in a "structure graph" are coupled by a submodular function, and we call the resulting problem *minimum cooperative cut*:

Definition 1 (Minimum Cooperative Cut)**.** *Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a non-decreasing submodular function $f : 2^{\mathcal{E}} \to \mathbb{R}_+$ defined on subsets of* **edges** *$\mathcal{E}$, find an (s,t)-cut $\Gamma \subseteq \mathcal{E}$ having minimum cost $f(\Gamma)$.*

A set function $f : 2^{\mathcal{E}} \to \mathbb{R}$ is submodular if it satisfies *diminishing marginal costs*: for any $A \subseteq B \subseteq \mathcal{E} \setminus \{e\}$, it holds that $f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B)$. Marginal costs are defined as $\rho_e(A) = f(A \cup \{e\}) - f(A)$. The standard cost function, a sum of edge weights, has constant marginal costs and is thus a *modular* function. The function is nondecreasing if $A \subseteq B$ implies that $f(A) \leq f(B)$.

We build on the well-known equivalence between binary labelings and graph cuts that we briefly sketch here. We aim to infer the value of n random variables that take values in a discrete label space $\mathcal{L}$. In particular, we seek the MAP labeling for the posterior distribution $p(\mathbf{x}|\mathbf{z}) \propto \exp(-E(\mathbf{x};\mathbf{z}))$, given an observation $\mathbf{z}$. For binary labels and an energy E restricted to pairwise submodular potentials, $E(\mathbf{x}) = \sum_{i=1}^{n} \psi_i(x_i) + \sum_{(i,j)\in\mathcal{N}} \psi_{ij}(x_i, x_j)$, energy minimization is equivalent to minimum cut. Construct a weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ that has a node v_i for each x_i, and two extra terminal nodes s and t, connected to each v_i by edges in $\mathcal{E}_t \subset \mathcal{E}$. The remaining edges, $\mathcal{E}_n$, connect the v_i. A minimal[1] (s,t)-cut $\Gamma \subseteq \mathcal{E}$ includes either edge (v_i, t), assigning v_i to s, or edge (s, v_i), and thus induces a binary labeling. Equivalently, each labeling $\mathbf{x}$ induces a cut $\Gamma(\mathbf{x})$. Let $X_s(\mathbf{x}) = \{v_i \in \mathcal{V} | x_i = 1\} \cup \{s\}$. Then the cut is $\Gamma(\mathbf{x}) = \Gamma(X_s(\mathbf{x})) = \{e = (u,v) \in \mathcal{E} | u \in X_s(\mathbf{x}), v \notin X_s(\mathbf{x})\}$. The graph $\mathcal{G}$ represents energy E if

$$E(\mathbf{x}) + \text{const} = \sum_{e\in\Gamma(\mathbf{x})\cap\mathcal{E}_t} w(e) + \sum_{e\in\Gamma(\mathbf{x})\cap\mathcal{E}_n} w(e) = w(\Gamma(\mathbf{x})). \quad (1)$$

We keep the weights on the terminal edges $\mathcal{E}_t$ that express the unary potentials ψ_i and replace the second sum by a cooperative cut (see also [5]):

$$E_f(\mathbf{x}) = \sum_{e\in\Gamma(\mathbf{x})\cap\mathcal{E}_t} w(e) + f(\Gamma(\mathbf{x}) \cap \mathcal{E}_n). \quad (2)$$

In the sequel, we will implicitly assume that f only operates on $\mathcal{E}_n$, and not explicitly write the restriction. The energy E_f is no longer submodular, and it is often global. In fact,

[1] A cut C is *minimal* if no proper subset $B \subset C$ is a cut. A *minimum* cut is usually also a minimal cut.

minimum cooperative cut is NP-hard [4], but the graph and edge submodularity provide enough structure to allow for approximation algorithms [4, 5]. This has been used, *e.g.*, for image segmentation with global boundary features. We will now extend this binary energy to multiple labels.

2. Multi-label cooperative cut models

In the binary case, the cost of the cut is computed on the directed edges from X_s (label 1) to the nodes X_t connected to t (label 0). In the case of more than two labels, there are several possible generalizations. We mention two, M1 and M2. Given $\mathbf{x}$, denote by $X_k(\mathbf{x}) = \{v_i \in \mathcal{V} | x_i = k\}$ the set of nodes with label k, for each $k \in \mathcal{L}$. We assume that we are given a structure graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ whose nodes correspond to the variables x_i. In the sequel, the functions f and $\{f_k\}_k$ are all submodular. We also assume all ψ_i to be nonnegative (otherwise they can be shifted by a constant).

M1 We treat all labels alike. Here, $\mathcal{G}$ is an undirected graph, and we view a labeling as an undirected $|\mathcal{L}|$-cut $\Gamma(\mathbf{x}) = \Gamma(X_1, \ldots, X_{|\mathcal{L}|}) = \{e = (v_i, v_j) | x_i \neq x_j\}$. Then the energy is $E_f(\mathbf{x}) = \sum_i \psi_i(x_i) + f(\Gamma(\mathbf{x}))$.

M2 We separate labels, *i.e.*, the boundary for each label is independent of the other labels' boundaries. Let the boundary of label $k \in \mathcal{L}$ in a directed graph $\mathcal{G}$ be $\Gamma_k(\mathbf{x}) = \Gamma(X_k) = \{e = (v_i, v_j) | x_i = k, x_j \neq k\}$, and define

$$E_f(\mathbf{x}) = \sum_i \psi_i(x_i) + \sum_{k \in \mathcal{L}} f_k(\Gamma_k(\mathbf{x})). \qquad (3)$$

The functions f_k can be different from each other or identical, depending on the application.

Here, we focus on M2, but analogous results hold for M1. Note that M2 implicitly favors fewer labels: the joint cost of edges coupled by one f_k is lower than the sum of their individual costs, and coupling is only possible for edges that are in the boundary of the same label.

We can take two routes to optimize these energies. On the one hand, we can derive an adaptive approximation as in [5], which results in an approximate pairwise energy function E_h and leads to an iterative algorithm. We show this approximation in Section 3. Any algorithm for multi-label pairwise MRFs applies to minimize E_h. Alternatively, we can build on the cut formulation and move-making algorithms [2]. In Section 4 we show that the best expansion move can be found as a minimum cooperative cut. That means any approximation algorithm for cooperative cut finds a move with an approximation guarantee. In Section 5, we show one bound resulting from the expansion construction and the approximation in Section 3. This workshop paper can only summarize results; details and proofs are deferred to a full version of the paper.

3. Adaptive upper bounds

From the submodular cost functions f_k, we can derive upper bounds for an iterative minimization just as described in [5]. We use the same upper bounds:

Lemma 1. *(Lemma 1 in [5]) For a submodular $f : 2^{\mathcal{E}} \to \mathbb{R}_+$ and an arbitrary $C \subseteq \mathcal{E}$, define $h_{f,C} : 2^{\mathcal{E}} \to \mathbb{R}_+$ as*

$$h_{f,C}(A) \triangleq f(C) + \sum_{e \in A \setminus C} \rho_e(C) - \sum_{e \in C \setminus A} \rho_e(\mathcal{E} \setminus \{e\}). \quad (4)$$

The function $h_{f,C}$ is a modular upper bound on f.

Importantly, if f is replaced by h, the min-cut problem becomes a minimum cut with a sum-of-weights cost, and the corresponding energy E_h a pairwise potential. For M2, we derive an upper bound h_k for each f_k separately, and sum up the appropriate edge weights afterwards. Given an initial labeling $\mathbf{x}'$, let $\Gamma'_k = \Gamma(X_k(\mathbf{x}'))$ be the boundary for label $k \in \mathcal{L}$. We use such boundaries as reference sets, and use marginal costs $\rho^k_e(A) = f_k(A \cup \{e\}) - f_k(A)$. For a new $\Gamma_k = \Gamma(X_k(\mathbf{x}))$ (resulting from a changed labeling $\mathbf{x}$), we get the following bound:

$$\begin{aligned} f_k(\Gamma_k) &\leq f_k(\Gamma'_k) + \sum_{e \in \Gamma_k \setminus \Gamma'_k} \rho^k_e(\Gamma'_k) - \sum_{e \in \Gamma'_k \setminus \Gamma_k} \rho^k_e(\mathcal{E} \setminus \{e\}) \\ &= \underbrace{f_k(\Gamma'_k)}_{\text{const.}} + \sum_{e \in \Gamma_k \setminus \Gamma'_k} \underbrace{\rho^k_e(\Gamma'_k)}_{c_k(e)} - \underbrace{\sum_{e \in \Gamma'_k} \rho^k_e(\mathcal{E} \setminus \{e\})}_{\text{const.}} \\ &\quad + \sum_{e \in \Gamma'_k \cap \Gamma_k} \underbrace{\rho^k_e(\mathcal{E} \setminus \{e\})}_{c_k(e)} = h_k(\Gamma_k). \end{aligned}$$

The bound essentially consists of edge weights $c_k(e) = \rho^k_e(\mathcal{E} \setminus \{e\})$ for $e \in \Gamma'_k$, and $c_k(e) = \rho^k_e(\Gamma'_k)$ for $e \notin \Gamma'_k$. The complete approximation of E_f is

$$E_h(\mathbf{x}) = \sum\nolimits_i \psi_i(x_i) + \sum\nolimits_k h_k(\Gamma_k(\mathbf{x})). \qquad (5)$$

The energy E_h consists of constants, unary terms and asymmetric pairwise potentials of the form

$$\psi_{ij}(x_i, x_j) = \begin{cases} 0 & \text{if } x_i = x_j, \\ c_k((v_i, v_j)) + c_\ell((v_j, v_i)) & \text{if } x_i = k, x_j = \ell \end{cases}$$

For optimization, we point out that $c_k(e) \geq 0$ for any $e \in \mathcal{E}_n$ when the f_k in question are monotone non-decreasing.

4. Cooperative α-expansions

Before we use the upper bound, we consider expansion moves for the non-pairwise cooperative energies E_f. Move-making algorithms [2] have become popular tools to extend graph cut-based optimization from binary to multiple labels.

In the sequel, we will focus on α-expansions, but $\alpha\beta$-swap moves are equally possible. Given a current labeling $\mathbf{x}'$, an expansion move with respect to a label α allows to switch any label $x'_i \neq \alpha$ to α, but any variable $x'_j = \alpha$ remains fixed. That means, if $\mathbf{x}$ is the labeling after the move, then $X_\alpha(\mathbf{x}') \subseteq X_\alpha(\mathbf{x})$. For certain energies, the expansion move that leads to the minimum possible energy (within the range of allowed moves) can be determined exactly by a graph cut [2]. Usual conditions are that the potentials are at most pairwise (or rephrased as pairwise), and that they are metric [2]. The energies E_f obviously do not satisfy those constraints, so we construct an auxiliary graph different from [2], and replace minimum cuts by minimum cooperative cuts. Since sums of weights are submodular as well, the construction also holds for asymmetric pairwise potentials if $\psi_{ij}(x_i, x_j) \geq 0$ and $\psi_{ij}(x_i, x_j) = 0$ for $x_i = x_j$.

We next show that the best expansion move corresponds to a minimum cooperative cut.

Lemma 2. *Computing the best α-expansion move with respect to the energy E_f (in M2) and any given labeling $\mathbf{x}'$ is a minimum cooperative cut.*

To constructively prove Lemma 2, we construct a directed graph $\widetilde{\mathcal{G}}_\alpha = (\widetilde{\mathcal{V}}, \widetilde{\mathcal{E}})$ and show two claims. The Lemma then follows as a corollary.

Claim 1. *A minimal cut in $\widetilde{\mathcal{G}}_\alpha$ corresponds to a labeling that is within one α-expansion of the current labeling $\mathbf{x}'$.*

Claim 2. *The cost of any such cut is equal to the energy of the associated labeling.*

The nodes $\tilde{v}_i \in \widetilde{\mathcal{V}}$ in $\widetilde{\mathcal{G}}_\alpha$ are in one-to-one correspondence with nodes $v_i \in \mathcal{V}$ in the given $\mathcal{G}$, and $\widetilde{\mathcal{G}}_\alpha$ has two terminal nodes $\tilde{s}, \tilde{t}$ that are connected to each $\tilde{v}_i$. Formally, a property similar to Properties 4.2, 5.2 in [2] holds for $\widetilde{\mathcal{G}}_\alpha$.

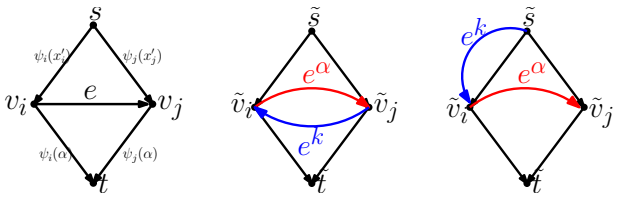
Property 1. *For a minimal cut $\widetilde{\Gamma}$ in $\widetilde{\mathcal{G}}_\alpha$, the following holds:*

1. *if $(s, v_i), (s, v_j) \in \widetilde{\Gamma}$, then $(v_i, v_j), (v_j, v_i) \notin \widetilde{\Gamma}$;*
2. *if $(v_i, t), (v_j, t) \in \widetilde{\Gamma}$, then $(v_i, v_j), (v_j, v_i) \notin \widetilde{\Gamma}$;*
3. *if $(s, v_i), (v_j, t) \in \widetilde{\Gamma}$, then $\tilde{\mathcal{E}} \cap \{(v_j, v_i)\} \subseteq \widetilde{\Gamma}$, $(v_i, v_j) \notin \widetilde{\Gamma}$;*
4. *for each v_i, either $(s, v_i) \in \widetilde{\Gamma}$ or $(v_i, t) \in \widetilde{\Gamma}$.*

Only one terminal edge is cut because the cut $\widetilde{\Gamma}$ is minimal. If the marginal costs remain strictly positive, then every minimum cut is also minimal. Otherwise, we add tiny additional weights to all edges.

The last point of Property 1 justifies the following labeling corresponding to a given cut $\widetilde{\Gamma}$:

$$x_i = \begin{cases} \alpha & \text{if } (\tilde{v}_i, t) \in \widetilde{\Gamma} \\ x'_i & \text{if } (s, \tilde{v}_i) \in \widetilde{\Gamma}. \end{cases} \tag{6}$$



(a) $\mathcal{G}$ (b) $x'_i = x'_j = k$ (c) $x'_i = k, x'_j = \ell$

Figure 1. (a) Edge e in $\mathcal{G}$; (b),(c) corresponding edges in $\widetilde{\mathcal{G}}_\alpha$ for $k, \ell \neq \alpha$. Colors indicate S_α and S_k.

We construct $\widetilde{\mathcal{G}}_\alpha$ to possess terminal edges that carry the ψ_i: $(\tilde{s}, \tilde{v}_i)$ with weight $\psi_i(x'_i)$ and $(\tilde{v}_i, t)$ with weight $\psi_i(x_i = \alpha)$. If $x'_i = \alpha$, then edge $(\tilde{s}, \tilde{v}_i)$ has weight ∞, so that $v_i \in X_{\tilde{s}}$ $(x_i = \alpha)$ is ensured. Then Labeling (6) is obviously within one α-expansion of $\mathbf{x}'$; this proves Claim 1.

For Claim 2, we need to detail the structure of $\widetilde{\mathcal{G}}_\alpha$. The construction of the remaining edges $\tilde{\mathcal{E}}_c$ in $\widetilde{\mathcal{G}}_\alpha$ is somewhat complicated because a directed cut only includes the edges between the $\tilde{s}$-part (label α) and $\tilde{t}$-part (label $k \neq \alpha$), but we must keep track of the cost of all boundaries, not only $\Gamma_\alpha(\mathbf{x})$. Thus, the cost of these other edges must be transferred appropriately. This is achieved by mapping edges in $\tilde{\mathcal{E}}_c$ to $\mathcal{E}_n$ in $\mathcal{G}$ as $\pi(\tilde{e}) \in \mathcal{E}_n$ (as opposed to distributing weights as in the sum-of-weights case). In addition, the edges $\tilde{\mathcal{E}}_c$ make $\widetilde{\mathcal{G}}_\alpha$ a multi-graph, and are partitioned into $(S_1, \ldots, S_{|\mathcal{L}|})$, $S_k \subseteq \tilde{\mathcal{E}}_c$. The cost of a set of edges $\widetilde{\Gamma} \subseteq \tilde{\mathcal{E}}_c$ is defined by a submodular function $\tilde{f}$ that uses the f_k on the mappings $\pi(\widetilde{\Gamma})$ of edges in $\widetilde{\Gamma}$. The S_k determine which f_k is used.

$$\tilde{f}(\widetilde{\Gamma}) = \sum_{k \in \mathcal{L}} f_k(\pi(\widetilde{\Gamma} \cap S_k)). \tag{7}$$

For any $e = (v_i, v_j) \in \mathcal{E}_n$ in $\mathcal{G}$, there are the following cases to construct edges in $\widetilde{\mathcal{G}}_\alpha$. The resulting edges form $\tilde{\mathcal{E}}_c$, and we indicate $\tilde{e} \in S_k$ by a superscript k on $\pi(\tilde{e})$.

$x'_i = \alpha, x'_j = k:$ e can at most be in $\Gamma_\alpha(\mathbf{x})$; introduce an edge $\tilde{e} = (\tilde{v}_i, \tilde{v}_j) \in S_\alpha$, with $\pi(\tilde{e}) = e^\alpha$;

$x'_i = k, x'_j = \alpha:$ this direction is never cut in $\widetilde{\mathcal{G}}_\alpha$, so introduce $\tilde{e} = (\tilde{v}_j, \tilde{v}_i) \in S_k$, with $\pi(\tilde{e}) = e^k$;

$x'_i = k, x'_j = \ell:$ e is in $\Gamma_k(\mathbf{x})$ if $x_i = x'_i = k$, and otherwise possibly in $\Gamma_\alpha(\mathbf{x})$; introduce $\tilde{e} = (\tilde{v}_i, \tilde{v}_j) \in S_\alpha$ with $\pi(\tilde{e}) = e^\alpha$, and an edge $\tilde{e}' = (\tilde{s}, \tilde{v}_i) \in S_k$ with $\pi(\tilde{e}') = e^k$;

$x'_i = k, x'_j = k:$ e is currently in no boundary; introduce an edge $\tilde{e} = (\tilde{v}_i, \tilde{v}_j) \in S_\alpha$, with $\pi(\tilde{e}) = e^\alpha$, and an edge $\tilde{e}' = (\tilde{v}_j, \tilde{v}_i) \in S_k$, with $\pi(\tilde{e}') = e^k$;

$x'_i = \alpha, x'_j = \alpha:$ labels are fixed, so we introduce no edges.

Figure 1 shows examples, and Table 1 lists all edges between two given nodes in the resulting $\widetilde{\mathcal{G}}_\alpha$.

x'_i	x'_j	$\pi(\tilde{e})$ of $(\tilde{v}_i, \tilde{v}_j)$ in $\widetilde{\mathcal{G}}_\alpha$	w_h
α	k	$(v_i, v_j)^\alpha$ for $\tilde{e} \in S_\alpha$, $(v_j, v_i)^k$ for $\tilde{e} \in S_k$	$\rho^\alpha_{(v_i,v_j)}(\mathcal{E} \setminus (v_i, v_j))$ $+\rho^k_{(v_j,v_i)}(\mathcal{E} \setminus (v_j, v_i))$
k	ℓ	$(v_i, v_j)^\alpha$ for $\tilde{e} \in S_\alpha$	$\rho^\alpha_{(v_i,v_j)}(\Gamma'_\alpha)$
k	k	$(v_i, v_j)^\alpha$ for $\tilde{e} \in S_\alpha$, $(v_j, v_i)^k$ for $\tilde{e} \in S_k$	$\rho^\alpha_{(v_i,v_j)}(\Gamma'_\alpha)$ $+\rho^k_{(v_j,v_i)}(\Gamma'_k)$

terminal edges in $\widetilde{\mathcal{G}}_\alpha$			
edges	x'_i	$\pi(\tilde{e})$	w_h
$(\tilde{s}, \tilde{v}_i)$	α	$\emptyset$	∞
$(\tilde{s}, \tilde{v}_i)$	k	T_i	$\psi_i(k) + \sum_{e \in \mathcal{N}_{i,k}} \rho^k_e(\mathcal{E} \setminus e)$
$(\tilde{v}_i, \tilde{t})$	$x'_i \in \mathcal{L}$	$\emptyset$	$\psi_i(\alpha)$

Table 1. Mappings $\pi(\tilde{e})$ of edges $\tilde{e}$ in $\widetilde{\mathcal{G}}_\alpha$ (if $x'_j = \alpha$, then there are no edges $(\tilde{v}_i, \tilde{v}_j)$). If (v_i, v_j) or $(v_i, v_j) \notin \mathcal{E}$, then replace it by $\emptyset$ in the table. The weights correspond to the adaptive upper bounds with respect to $\mathbf{x}'$ after collapsing parallel edges and summing their approximate costs c_k. Here, $\mathcal{N}_{i,k} = \{e = (v_i, v_j) \in \mathcal{E} | x'_i = k, x'_j \neq k, \alpha\}$ and $T_i = \bigcup_{e \in \mathcal{N}_{i,k}} \{\, e^k \,\}$.

Given this $\widetilde{\mathcal{G}}_\alpha$, we now show that $E_f(\mathbf{x}) = \tilde{f}(\widetilde{\Gamma}(\mathbf{x}) \cap \tilde{\mathcal{E}}_c) + \sum_i \psi_i(x_i)$. To do so, we prove that $\pi(\widetilde{\Gamma}(\mathbf{x}) \cap S_k) = \Gamma_k(\mathbf{x})$ for all labels $k \in \mathcal{L}$.

Let $\widetilde{\Gamma} \subseteq \tilde{\mathcal{E}}$ by a minimal (s,t)-cut in $\widetilde{\mathcal{G}}_\alpha$, and $\mathbf{x}$ the corresponding labeling (6), and $\Gamma_k(\mathbf{x})$ the boundary of label k in $\mathcal{G}$. Then Table 1 implies for $k \neq \alpha$ that

$$
\begin{aligned}
&\pi(\widetilde{\Gamma} \cap S_k) \\
&= \Big(\bigcup_{\substack{(\tilde{v}_i, \tilde{v}_j) \in S_k \\ x_i = \alpha, x_j \neq \alpha}} \pi(\tilde{v}_i, \tilde{v}_j) \Big) \cup \Big(\bigcup_{(\tilde{s}, \tilde{v}_i) \in \widetilde{\Gamma} \cap S_k} \pi(\tilde{s}, \tilde{v}_i) \Big) \\
&= \Big(\bigcup_{\substack{(v_j, v_i) \in \mathcal{E}: \\ x'_i = \alpha, \\ x_j = x'_j = k}} (v_j, v_i) \Big) \cup \Big(\bigcup_{\substack{(v_i, v_j) \in \mathcal{E}: \\ x_i = \alpha, \\ x'_i = x_j = k}} (v_j, v_i) \Big) \\
&\quad \cup \Big(\bigcup_{x_i = k} \bigcup_{\substack{j : (v_i, v_j) \in \mathcal{E}: \\ x_j \neq k, \alpha}} (v_i, v_j) \Big) \\
&= \big(\Gamma_k \cap (X_k \times X_\alpha) \big) \cup \Big(\Gamma_k \cap \bigcup_{\ell \neq k, \alpha} (X_k \times X_\ell) \Big) = \Gamma_k(\mathbf{x}).
\end{aligned}
$$

Similarly, one can show that $\pi(\widetilde{\Gamma} \cap S_\alpha) = \Gamma_\alpha(\mathbf{x})$. In consequence, $\tilde{f}(\widetilde{\Gamma}(\mathbf{x})) = \sum_k f_k(\Gamma_k(\mathbf{x}))$. The expression of the unary terms is analogous in $\mathcal{G}$ and $\widetilde{\mathcal{G}}_\alpha$, so indeed $E_f(\mathbf{x}) = \tilde{f}(\widetilde{\Gamma}(\mathbf{x}) \cap \tilde{\mathcal{E}}_c) + \sum_i \psi_i(x_i)$. This proves Claim 2.

5. Approximation factor

One simple way to solve the cut for the expansion moves above is to approximate $\tilde{f}$ as in Section 3 — Table 1 shows the resulting edge weights, if all parallel edges are collapsed into one edge, and their weights c_k summed up to w_h. There are many ways to then interleave expansion moves and adaptations of the upper bound. Here, we state an approximation factor for computing an upper bound $h_{\tilde{f},C}$ with respect to one current set $C = \Gamma'$, and then running α-expansions until convergence (alternating α as in [2]), while keeping the cost $h_{\tilde{f},C}$ fixed.

Lemma 3. *Let $\hat{x}$ be a local minimum reached via a sequence of α-expansions (until convergence) with cost $h_{\tilde{f},C}$, and let $\mathbf{x}^C = \operatorname{argmin}_{\mathbf{x} \in \{0,1\}^n} E_{h_{\tilde{f},C}}(\Gamma(\mathbf{x}))$. Then*

$$E_{h_{f,C}}(\Gamma(\hat{\mathbf{x}})) \leq (1 + \gamma) E_{h_{f,C}}(\Gamma(\mathbf{x}^C)),$$

where $\gamma = \max_{\tilde{e} \in \tilde{\mathcal{E}}, k, \ell \in \mathcal{L}} h_k(e)/h_\ell(e) \leq \max_{e,k,\ell} \rho^k_e(C)/\rho^\ell_e(\mathcal{E} \setminus \{e\}) \leq \max_{e,k,\ell} f_k(e)/\rho^\ell_e(\mathcal{E} \setminus \{e\})$.

The proof of the Lemma is technical and relies on the same strategy as the proof of Theorem 6.1 in [2].

Next we derive a bound on the solution of an iterative algorithm that starts with the α-expansion for $h_{\tilde{f},\emptyset}$ (until convergence), and then continues with α-expansions for $h_{\tilde{f},C}$, where C is the best solution from the previous series of expansions. Let $\hat{\mathbf{x}}$ be the best solution achieved this way, and let $\mathbf{x}^*$ be the global optimum for E_f.

Lemma 4. *Let β be such that $h_{\tilde{f},\emptyset}(\Gamma(\mathbf{x}^*)) \leq \beta f(\Gamma(\mathbf{x}^*))$. Then*

$$E_f(\hat{\mathbf{x}}) \leq \beta(1 + \gamma) E_f(\mathbf{x}^*),$$

where γ is as in Lemma 3.

The factor β is discussed in [5]. The proof relies on Lemma 3, the properties of h, and the optimality of $\hat{x}$.

References

[1] Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. In *ICCV*, 2001.

[2] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE TPAMI*, 23, 2001.

[3] H. Ishikawa. Higher-order clique reduction in binary cut. In *CVPR*, 2009.

[4] S. Jegelka and J. Bilmes. Cooperative cuts: graph cuts with submodular edge weights. Technical Report TR-189, Max Planck Institute for Biological Cybernetics, 2010.

[5] S. Jegelka and J. Bilmes. Submodularity beyond submodular energies: coupling edges in graph cuts. In *CVPR*, 2011.

[6] P. Kohli and M. Kumar. Energy minimization for linear envelope MRFs. In *CVPR*, 2010.

[7] P. Kohli, M. P. Kumar, and P. Torr. P3 & beyond: solving energies with higher-order cliques. In *CVPR*, 2007.

[8] V. Kolmogorov and C. Rother. Minimizing nonsubmodular functions with graph cuts–a review. *IEEE TPAMI*, 29(7):1274–1279, 2007.

[9] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE TPAMI*, 26(2):147–159, 2004.

[10] N. Komodakis and N. Paragios. Beyond pairwise energies: efficient optimization for higher-order MRFs. In *CVPR*, 2009.