Shekhovtsov, A. and Hlavac, V. On partial optimality by auxiliary submodular problems. *Control Systems and Computers*, (2), 2011.

Shekhovtsov, A., Swoboda, P., and Savchynskyy, B. Maximum persistency via iterative relaxed inference in graphical models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(7):1668–1682, 2017.

Shotton, J., Winn, J., Rother, C., and Criminisi, A. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *European conference on computer vision*, pp. 1–15. Springer, 2006.

Sontag, D. and Jaakkola, T. S. New outer bounds on the marginal polytope. In *Advances in Neural Information Processing Systems*, pp. 1393–1400, 2008.

Sontag, D. A. *Approximate inference in graphical models using LP relaxations*. PhD thesis, Massachusetts Institute of Technology, 2010.

Swoboda, P., Shekhovtsov, A., Kappes, J. H., Schnörr, C., and Savchynskyy, B. Partial optimality by pruning for map-inference with general graphical models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(7), 2016.

Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M., and Rother, C. 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):1068–1080, 2008.

Tappen and Freeman. Comparison of graph cuts with belief propagation for stereo, using identical mrf parameters. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pp. 900–906 vol.2, 2003.

Wainwright, M. J. and Jordan, M. I. *Graphical models, exponential families, and variational inference*. Now Publishers Inc, 2008.

Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., and Torr, P. H. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 1529–1537, 2015.

Graph cuts always find a global optimum for Potts models (with a catch): supplementary material

A. Proof of Theorem 3

In this section, we give the full proof of Theorem 3, restated here. Theorem 2 is then a straightforward corollary of Theorem 3 (Theorem 3 is essentially the constructive version of Theorem 2).

Theorem. *Consider an input instance θ with Potts pairwise potentials and weights w, and let the labeling x be a local minimum for θ with respect to expansion moves. Define perturbed weights $w^x : E \to \mathbb{R}_+$ as*

$$w^x_{uv} = \begin{cases} w_{uv} & x(u) \neq x(v) \\ 2w_{uv} & x(u) = x(v), \end{cases} \tag{12}$$

and let

$$\theta^x_{uv}(i,j) = w^x_{uv}\mathbb{I}[i \neq j] \tag{13}$$

be the pairwise Potts energies corresponding to the weights w^x. Then x is a global minimum in the instance with objective vector $\theta^x = (\theta_u : u;\ \theta^x_{uv} : uv)$. This is the Potts model instance with the same node costs $\theta_u(i)$ as the original instance, but new pairwise energies $\theta^x_{uv}(i,j)$ defined using the perturbed weights w^x. Additionally, the LP relaxation (2) is tight on this perturbed instance.

Proof. Let x be any labeling of G. We show that there exists an expansion x^α of x such that for some $\epsilon > 0$:

$$\langle \theta, x - x^\alpha \rangle \geq \epsilon \left(\langle \theta^x, x \rangle - \min_{y \in L(G)} \langle \theta^x, y \rangle \right). \tag{14}$$

This implies that as long as $\langle \theta^x, x \rangle - \min_{y \in L(G)} \langle \theta^x, y \rangle$ is positive, there exists an expansion move with strictly better objective than x. The right-hand-side of (14) is always nonnegative, because $x \in L(G)$. Therefore, x can only be a local minimum w.r.t. expansion moves if $\langle \theta^x, x \rangle = \min_{y \in L(G)} \langle \theta^x, y \rangle$. Every labeling of G corresponds to a point in $L(G)$, since $M(G) \subset L(G)$, so if $\langle \theta^x, x \rangle = \min_{y \in L(G)} \langle \theta^x, y \rangle$, x must be an optimal labeling in the instance with objective θ^x. This equality also implies that a vertex of $M(G)$ attains the optimal objective value for (2), which is the definition of “tightness” on an instance. So (14) dgives both claims of the theorem.

Let $y' \in \arg\min_{y \in L(G)} \langle \theta^x, y \rangle$ be an LP solution to the perturbed instance. To show (14), we design a rounding algorithm R that takes y' and x as input and outputs an expansion move x^α of x. We show that R satisfies

$$\mathbb{E}[\langle \theta, x - R(x, y') \rangle] \geq \epsilon(\langle \theta^x, x - y' \rangle), \tag{15}$$

which proves (14) because it implies there exists some x^α in the support of $R(x, y')$ that attains (14).

Algorithm 2 $R(x, y')$

```
1: Fix 0 < ϵ < 1/k.
2: Set x' = ϵy' + (1 − ϵ)x.
3: Choose i ∈ {1, . . . , k} uniformly at random.
4: Choose r ∈ (0, 1/k) uniformly at random.
5: Initialize labeling x^α : V → [k].
6: for each u ∈ V do
7:     if x'_u(i) > r then
8:         Set x^α(u) = i.
9:     else
10:        Set x^α(u) = x(u)
11:    end if
12: end for
13: Return x^α
```

Lemma 2 (Rounding guarantees)**.** *The labeling x^α output by Algorithm 2 is an expansion of x, and it satisfies the following guarantees:*

$$\begin{aligned} &\mathbb{P}[x^\alpha(u) = i] = x'_u(i) && \forall\, u \in V, i \in [k] \\ &\mathbb{P}[x^\alpha(u) \neq x^\alpha(v)] \leq 2d(u, v) && \forall\, (u, v) \in E : x(u) = x(v) \\ &\mathbb{P}[x^\alpha(u) = x^\alpha(v)] = (1 - d(u, v)) && \forall\, (u, v) \in E : x(u) \neq x(v), \end{aligned}$$

where $d(u, v) = \frac{1}{2} \sum_i |x'_u(i) - x'_v(i)|$.

Proof of Lemma 2 (rounding guarantees). The output x^α is clearly an i-expansion of x for the i chosen in line 3.

For the guarantees, first, fix $u \in V$ and a label $i \neq x(u)$. We output $x^\alpha(u) = i$ precisely when i is chosen in line 3, and $0 < r < x'_u(i)$, which occurs with probability $\frac{1}{k} \frac{x'_u(i)}{1/k} = x'_u(i)$ (we used here that $x'_u(i) \leq \epsilon < 1/k$ for all $i \neq x(u)$). Now we output $x^\alpha(u) = x(u)$ with probability $1 - \sum_{j \neq x(u)} \mathbb{P}[x^\alpha(u) = j] = 1 - \sum_{j \neq x(u)} x'_u(j) = x'_u(x(u))$, since $\sum_i x'_u(i) = 1$. This proves the first guarantee.

For the second, consider an edge (u, v) not cut by x, so $x(u) = x(v)$. Then (u, v) is cut by x^α when some $i \neq x(u)$ is chosen and r falls between $x'_u(i)$ and $x'_v(i)$. This occurs with probability

$$\frac{1}{k} \sum_{i \neq x(u)} \frac{\max(x'_u(i), x'_v(i)) - \min(x'_u(i), x'_v(i))}{1/k} = \sum_{i \neq x(u)} |x'_u(i) - x'_v(i)| \leq 2d(u, v).$$

Finally, consider an edge (u, v) cut by x, so that $x(u) \neq x(v)$. Here $x^\alpha(u) = x^\alpha(v)$ if some i, r are chosen with $r < \min(x'_u(i), x'_v(i))$. We have $r < \min(x'_u(i), x'_v(i))$ with probability $\frac{\min(x'_u(i), x'_v(i))}{1/k}$. Note that this is still valid if $i = x(u)$ or $i = x(v)$, since only one of those equalities can hold. So we get

$$\mathbb{P}[x^\alpha(u) = x^\alpha(v)] = \frac{1}{k} \sum_i \frac{\min(x'_u(i), x'_v(i))}{1/k} = \frac{1}{2} \left(\sum_i x'_u(i) + x'_v(i) - |x'_u(i) - x'_v(i)| \right) = 1 - d(u, v),$$

where we used again that $\sum_i x'_u(i) = 1$. $\square$

Algorithm 2 is very similar to the rounding algorithm from Lang et al. (2018), essentially just using different constants to give a simplified analysis. The algorithm used in Lang et al. (2018) was itself a simple modification of the ϵ-close rounding from Angelidakis et al. (2017).

With these guarantees in hand, we can now prove (15). Let $x^\alpha = R(x, y')$. Let $E^x = \{(u, v) \in E : x(u) \neq x(v)\}$ be the set of edges cut by x. Recall that $\theta_{uv}(i,j) = w_{uv}\mathbb{I}[i \neq j]$. Then we have:

$$\mathbb{E}[\langle \theta, x - x^\alpha \rangle] = \sum_u \theta_u(x(u))\mathbb{P}[x^\alpha(u) \neq x(u)] - \sum_u \sum_{i \neq x(u)} \theta_u(i)\mathbb{P}[x^\alpha(u) = i] + \sum_{uv \in E^x} w_{uv}\mathbb{P}[x^\alpha(u) = x^\alpha(v)] - \sum_{uv \in E \setminus E^x} w_{uv}\mathbb{P}[x^\alpha(u) \neq x^\alpha(v)].$$

Applying Lemma 2, we obtain:

$$\begin{aligned}\mathbb{E}[\langle \theta, x - x^\alpha \rangle] &\geq \sum_u \theta_u(x(u))(1 - x'_u(x(u))) - \sum_u \sum_{i \neq x(u)} \theta_u(i)x'_u(i) + \sum_{uv \in E^x} w_{uv}(1 - d(u,v)) - \sum_{uv \in E \setminus E^x} 2w_{uv}d(u,v) \\ &= \sum_u \theta_u(x(u)) + \sum_{uv \in E^x} w^x_{uv} - \sum_u \sum_i \theta_u(i)x'_u(i) - \sum_{uv \in E} w^x_{uv}d(u,v). \end{aligned} \quad (16)$$

Here we are using the formula for w^x_{uv} given by (17): $w^x_{uv} = w_{uv}$ if (u,v) is in E^x, and $2w_{uv}$ otherwise.

Because x is a vertex of $M(G)$, the node variables $x_u(i)$ are either 0 or 1. Then there is only one setting of $x_{uv}(i,j)$ that satisfies the marginalization constraints. So the edge cost paid by x on each edge is proportional to $\frac{1}{2}\sum_{uv} |x_u(i) - x_v(i)|$, since this is 1 if x labels u and v differently and 0 otherwise. Therefore,

$$\sum_{uv}\sum_{i,j} \theta^x_{uv}(i,j)x_{uv}(i,j) = \sum_{uv} \frac{w^x_{uv}}{2} \sum_i |x_u(i) - x_v(i)|.$$

The following proposition says we can also rewrite the edge cost paid by the LP solution y' in this way.

Proposition. *In uniform metric labeling, there is a closed form for the optimal edge cost that only involves the node variables. That is, fix arbitrary node variables $z_u(i)$ and $z_v(j)$. Then the value of*

$$\begin{aligned} \min_{z_{uv}} \quad & \sum_{i,j} \mathbb{I}[i \neq j] z_{uv}(i,j) \\ \textit{subject to} \quad & \sum_j z_{uv}(i,j) = z_u(i) \\ & \sum_i z_{uv}(i,j) = z_v(j) \\ & z_{uv}(i,j) \geq 0 \end{aligned}$$

is equal to $\frac{1}{2}\sum_i |z_u(i) - z_v(i)|$ (Archer et al., 2004; Lang et al., 2018). This fact is used to prove that the local LP relaxation is equivalent to the metric *LP relaxation for uniform metric labeling (Archer et al., 2004; Lang et al., 2018).*

Because y' is an optimal solution to (2) for objective θ^x, y' pays the minimum edge cost consistent with its node variables, since otherwise it cannot be optimal. Then the above proposition implies that:

$$\sum_{uv}\sum_{i,j} \theta^x_{uv}(i,j)y'_{uv}(i,j) = \sum_{uv} \frac{w^x_{uv}}{2} \sum_i |y'_u(i) - y'_v(i)|.$$

Since $x'_{uv}(i,j) = \epsilon y'_{uv}(i,j) + (1-\epsilon)x_{uv}(i,j)$, and $d(u,v)$ is convex,

$$d(u,v) = \frac{1}{2}\sum_i |x'_u(i) - x'_v(i)| \leq \frac{\epsilon}{2}\sum_i |y'_u(i) - y'_v(i)| + \frac{1-\epsilon}{2}\sum_i |x_u(i) - x_v(i)|$$

Using this, the definition of x', and the closed forms for the edge cost of y' and x, we can simplify (16) to:

$$\begin{aligned}\mathbb{E}[\langle\theta^x, x - x^\alpha\rangle] \geq \langle\theta^x, x\rangle - \Bigg[&(1-\epsilon)\sum_u \theta_u(x(u)) + \epsilon\sum_u\sum_i \theta_u(i)y'_u(i) + (1-\epsilon)\sum_{uv}\frac{w^x_{uv}}{2}\sum_i |x_u(i) - x_v(i)| \\ &+ \epsilon\sum_{uv}\frac{w^x_{uv}}{2}\sum_i |y_u(i) - y_v(i)|\Bigg] \\ &= \langle\theta^x, x\rangle - [(1-\epsilon)\langle\theta^x, x\rangle + \epsilon\langle\theta^x, y'\rangle] \\ &= \epsilon\langle\theta^x, x - y'\rangle,\end{aligned}$$

which is what we wanted to show. This analysis implies that for any expansion minimum x, (i) x is a MAP solution to the instance θ^x and (ii) the local LP relaxation (2) is tight on the instance θ^x. Point (ii) is crucial for the correctness of our algorithm in Section 5. However, in the next section we give a simpler proof of (i) that does not use the local LP relaxation. □

A.1. Combinatorial proof of Theorem 3 part (i)

Here, we give a simpler proof for the first claim of Theorem 3, that a solution x returned by α-expansion is the optimal labeling in the instance with objective θ^x. However, the extra guarantee of Theorem 3, that the local LP relaxation is tight on the instance with objective θ^x, was crucial to the correctness of our algorithm in Section 5.

Theorem. *Consider an input instance θ with Potts pairwise potentials and weights w, and let the labeling x be a local minimum for θ with respect to expansion moves. Define perturbed weights $w^x : E \to \mathbb{R}_+$ as*

$$w^x_{uv} = \begin{cases} w_{uv} & x(u) \neq x(v) \\ 2w_{uv} & x(u) = x(v), \end{cases} \tag{17}$$

and let

$$\theta^x_{uv}(i,j) = w^x_{uv}\mathbb{I}[i \neq j] \tag{18}$$

be the pairwise Potts energies corresponding to the weights w^x. Then x is a global minimum in the instance with objective vector $\theta^x = (\theta_u : u;\ \theta^x_{uv} : uv)$. This is the Potts model instance with the same node costs $\theta_u(i)$ as the original instance, but new pairwise energies $\theta^x_{uv}(i,j)$ defined using the perturbed weights w^x.

Proof. We'll show that if some assignment y obtains $\langle\theta^x, y\rangle < \langle\theta^x, x\rangle$, there exists an expansion move x^α of x with $\langle\theta, x^\alpha\rangle < \langle\theta, x\rangle$. Consequently, when x is optimal with respect to expansion moves, it is also the global optimal assignment in the instance with objective θ^x. Assume such a y exists and define $V^\alpha = \{u \in V | y(u) = \alpha\}$. This is the set of points labeled α by y. The sets $(V^1, \ldots, V^k)$ form a partition of V. For each $\alpha \in [k]$, define the expansion x^α of x towards y as:

$$x^\alpha(u) = \begin{cases} \alpha & u \in V^\alpha \\ x(u) & \text{otherwise.} \end{cases}$$

We will show:

$$\sum_\alpha (\langle\theta, x\rangle - \langle\theta, x^\alpha\rangle) \geq \langle\theta^x, x\rangle - \langle\theta^x, y\rangle \tag{19}$$

This immediately gives the result: if $\langle\theta^x, y\rangle < \langle\theta^x, x\rangle$, then at least one term in the sum on the left-hand-side must be positive, and this corresponds to an expansion x^α of x with better objective in the original instance.

Consider a single term $\langle\theta, x\rangle - \langle\theta, x^\alpha\rangle$ on the left-hand-side of (19). The difference in node cost terms is precisely $\sum_{u\in V^\alpha}\theta_u(x(u)) - \theta_u(x^\alpha(u))$, since on all $v \in V \setminus V^\alpha$, $x^\alpha(v) = x(v)$. This is equal to $\sum_{u\in V^\alpha}\theta_u(x(u)) - \theta_u(y(u))$, so the sum over α gives the difference in node cost between x and y:

$$\sum_\alpha\sum_{u\in V^\alpha}\theta_u(x(u)) - \theta_u(x^\alpha(u)) = \sum_{u\in V}\theta_u(x(u)) - \theta_u(y(u)). \tag{20}$$

For any assignment z, let $E_z \subset E$ be the set of edges (u, v) separated by z. Then we can write the difference in edge costs between x and x^α, with the original weights w_{uv}, as

$$\sum_{uv \in E_x \setminus E_{x^\alpha}} w_{uv} - \sum_{uv \in E_{x^\alpha} \setminus E_x} w_{uv},$$

and the edge cost difference between x and y with weights w^x as:

$$\sum_{uv \in E_x \setminus E_y} w_{uv} - \sum_{uv \in E_y \setminus E_x} 2w_{uv},$$

where we used the definition of w^x_{uv}. Then what remains is to show:

$$\sum_\alpha \left(\sum_{uv \in E_x \setminus E_{x^\alpha}} w_{uv} - \sum_{uv \in E_{x^\alpha} \setminus E_x} w_{uv} \right) \geq \sum_{uv \in E_x \setminus E_y} w_{uv} - \sum_{uv \in E_y \setminus E_x} 2w_{uv}.$$

Define B^α to be the set of edges with exactly one endpoint in V^α i.e., $B^\alpha = \{(u, v) \in E : |\{u, v\} \cap V^\alpha| = 1\}$. For all $(u, v) \in B^\alpha$, $y(u) \neq y(v)$, and either $y(u) = \alpha$ or $y(v) = \alpha$.

Let $(u, v) \in E_x \setminus E_y$. Because $y(u) = y(v)$, the edge (u, v) appears in *exactly one* of the $E_x \setminus E_{x^\alpha}$. That is, $y(u) = y(v) = \alpha$, so x^α does not cut (u, v), and x^β cuts (u, v) for all $\beta \neq \alpha$. This implies

$$\sum_\alpha \sum_{uv \in E_x \setminus E_{x^\alpha}} w_{uv} \geq \sum_{uv \in E_x \setminus E_y} w_{uv} \tag{21}$$

If x^α separates an edge (u, v) that is not separated by x, exactly one endpoint of (u, v) is in V^α, since otherwise both endpoints would have been assigned label α. Thus $E_{x^\alpha} \setminus E_x \subset B_\alpha \setminus E_x$. This implies

$$\sum_\alpha \sum_{uv \in E_{x^\alpha} \setminus E_x} w_{uv} = \sum_\alpha \sum_{uv \in B^\alpha \setminus E_x} w_{uv} = 2 \sum_{uv \in E_y \setminus E_x} w_{uv}, \tag{22}$$

where the last equality is because each edge in E_y appears in two B^α. Combining (21) and (22), we obtain:

$$\sum_\alpha \left(\sum_{uv \in E_x \setminus E_{x^\alpha}} w_{uv} - \sum_{uv \in E_{x^\alpha} \setminus E_x} w_{uv} \right) \geq \sum_{uv \in E_x \setminus E_y} w_{uv} - \sum_{uv \in E_y \setminus E_x} 2w_{uv}, \tag{23}$$

which is what we wanted. Combining (20) and (23), we obtain (19). $\square$

A.2. Proof of Lemma 1

Proof of lemma 1. Recall that $\mathcal{S}(\theta)$ is defined as the set of x for which there exists $\theta' \in \mathcal{I}(\theta)$ such that x is a MAP solution to the instance θ'. We want to show that $\mathcal{S}(\theta)$ can also be written as:

$$\mathcal{S}(\theta) = \{x : x \text{ a MAP solution to the instance } \theta^x \text{ defined by (4) and (5)}\}$$

To do show, we simply show that if x is a MAP solution for some $\theta' \in \mathcal{I}(\theta)$, then x is also the MAP solution to the instance θ^x. This is effectively because θ^x is the "best possible" perturbation for x that is contained in $\mathcal{I}(\theta)$. Fix $\theta' \in \mathcal{I}(\theta)$ for which x is a MAP solution. Then for all labelings $y \neq x$, $\langle \theta', y \rangle \geq \langle \theta', x \rangle$. In particular,

$$\sum_u \theta'_u(y(u)) + \sum_{uv} \theta'_{uv}(y(u), y(v)) \geq \sum_u \theta'_u(x(u)) + \sum_{uv} \theta'_{uv}(x(u), x(v)).$$

Because we assume throughout that $\theta_{uv}(i, j) = w_{uv}\mathbb{I}[i \neq j]$ (i.e., that the input instance is a Potts model), the definition of $\mathcal{I}(\theta)$ (equation 3) implies that every instance in $\mathcal{I}(\theta)$ is a Potts model. So let w' be the weights of the instance θ'. Additionally, recall that the definition of $\mathcal{I}(\theta)$ implies that $\theta'_u(i) = \theta_u(i)$ for all (u, i). Then the inequality above becomes:

$$\sum_u \theta_u(y(u)) - \sum_u \theta_u(x(u)) + \sum_{\substack{uv: y(u) \neq y(v) \\ x(u) = x(v)}} w'_{uv} - \sum_{\substack{uv: x(u) \neq x(v) \\ y(u) = y(v)}} w'_{uv} \geq 0$$

The definition of $\mathcal{I}(\theta)$ requires that for all (u, v), $w_{uv} \leq w'_{uv} \leq 2w_{uv}$. Together with the previous inequality, this implies

$$\sum_u \theta_u(y(u)) - \sum_u \theta_u(x(u)) + \sum_{\substack{uv: y(u) \neq y(v) \\ x(u) = x(v)}} 2w_{uv} - \sum_{\substack{uv: x(u) \neq x(v) \\ y(u) = y(v)}} w_{uv} \geq 0.$$

By definition of the perturbed weights w^x_{uv} (17), we have

$$\sum_u \theta_u(y(u)) - \sum_u \theta_u(x(u)) + \sum_{\substack{uv: y(u) \neq y(v) \\ x(u) = x(v)}} w^x_{uv} - \sum_{\substack{uv: x(u) \neq x(v) \\ y(u) = y(v)}} w^x_{uv} \geq 0,$$

which is equivalent to:

$$\langle \theta^x, y \rangle \geq \langle \theta^x, x \rangle.$$

Because y was arbitrary, this implies x is a MAP solution to the instance θ^x. $\square$

B. Comparing (7) and (11)

In this section, we expound on the relationship between (7) and (11), the bound obtained directly from α-expansion's objective approximation guarantee. In particular, we show that any x that is feasible for (7) is also feasible for (11). While we solve the relaxation (10) of (7) in practice, this gives some intuition for why (10) gives much tighter bounds than (11).

We have two ways of characterizing the set of labelings x that are local optima w.r.t. expansion moves. The first, guaranteed by Boykov et al. (2001), is that all such x satisfy

$$\langle \theta, x \rangle \leq \langle \theta, x^* \rangle + \sum_{uv \in E} w_{uv} \mathbb{I}[x^*(u) \neq x^*(v)], \tag{24}$$

where x^* is a MAP solution. That is, the "extra" objective paid by x is at most the edge cost paid by a MAP solution. The second, guaranteed by Theorem 2, is that x is the MAP solution in the instance with objective θ^x (i.e., $x \in \mathcal{S}(\theta)$). We now show that any labeling x that is a MAP solution in the instance with objective θ^x also satisfies (24), but the converse is not true. This implies that the feasible region of (7) is strictly smaller than that of (11).

Proposition. *Let x be a labeling that is optimal in the instance with objective θ^x, and let x^* be a MAP solution to the original instance, with objective θ. Then:*

$$\langle \theta, x \rangle \leq \langle \theta, x^* \rangle + \sum_{uv \in E} w_{uv} \mathbb{I}[x^*(u) \neq x^*(v)],$$

Proof. Because x is optimal for θ^x, we have $\langle \theta^x, x \rangle \leq \langle \theta^x, x^* \rangle$. Recall from the definitions of w^x_{uv} and $\theta^x_{uv}(i, j)$ ((17) and (18)) that $\langle \theta^x, x \rangle = \langle \theta, x \rangle$. We also have that

$$\begin{aligned}\langle \theta^x, x^* \rangle = \sum_u \theta_u(x^*(u)) + \sum_{uv} w^x_{uv} \mathbb{I}[x^*(u) \neq x^*(v)] &\leq \sum_u \theta_u(x^*(u)) + 2\sum_{uv} w_{uv} \mathbb{I}[x^*(u) \neq x^*(v)] \\ &= \langle \theta, x^* \rangle + \sum_{uv} w_{uv} \mathbb{I}[x^*(u) \neq x^*(v)].\end{aligned}$$

Here we used that $w^x_{uv} \leq 2w_{uv}$ for all $(u, v) \in E$. Therefore, $\langle \theta, x \rangle \leq \langle \theta, x^* \rangle + \sum_{uv \in E} w_{uv} \mathbb{I}[x^*(u) \neq x^*(v)]$. $\square$

Conversely, not all x satisfying (24) are optimal in the instance with objective θ^x. We now construct a simple example.

Example where (7) is much tighter than (11). Let $k = 4$ and consider a graph $G = (V, E)$ with two nodes s and t, and one edge (s, t). Let $w_{st} = 1$. For the node costs, Set $\theta_s(0) = 0$, $\theta_s(1) = \epsilon$, and $\theta_s(2) = \theta_s(3) = \infty$. Set $\theta_t(0) = \theta_t(1) = \infty$, $\theta_t(2) = \epsilon$, $\theta_t(3) = 0$. The MAP solution x^* clearly labels s with label 0 and t with label 3, for an objective of 1. Now consider the solution x that labels s with label 1 and t with label 2, for an objective of $1 + 2\epsilon$. For this x, because x cuts the only edge, $\theta^x = \theta$ (see (17)). Therefore, x is not optimal in the instance with objective θ^x, so it is not feasible for (7). However,

$$1 + 2\epsilon \leq \langle \theta, x^* \rangle + \sum_{uv \in E} w_{uv} \mathbb{I}[x^*(u) \neq x^*(v)] = 2$$

for $\epsilon < 1/2$. Therefore, x is feasible for (11). The Hamming distance between x and x^* is 1.0—x agrees with x^* on 0 nodes—so if we run (11) to bound the Hamming error on this instance, we obtain a bound of 1.0. On the other hand, (7) returns a Hamming distance bound of 0 for this instance, correctly indicating that α-expansion always recovers x^* regardless of the initialization. This is because any optimal x must cut (s, t), since otherwise it has infinite objective, and any x that cuts (s, t) has $\theta^x = \theta$, and x^* is the only optimal labeling for objective θ. Hence x^* is the only feasible point of (7) for this instance.

C. Model details

In this section, we give more details on the models used for our experiments in Section 6. These models are similar to the ones studied in Lang et al. (2019). There are two types of models: object segmentation and stereo vision.

C.1. Object segmentation

We use the object segmentation models from Shotton et al. (2006), which were also studied by Alahari et al. (2010) in the context of graph cut methods. These models are available as part of the OpenGM 2 benchmark (Kappes et al., 2015)[1]. In these models, G is a grid with one vertex per pixel and has edges connecting adjacent pixels. The node costs $\theta_u(i)$ are set based on a learned function of shape, color, and location features. Similarly, the edge weights are set using *contrast-sensitive* features:

$$w_{uv} = \eta_1 \exp\left(-\frac{||I(u) - I(v)||_2^2}{2\sum_{p,q}||I(p) - I(q)||_2^2}\right) + \eta_2,$$

where $\eta = (\eta_1, \eta_2)$, $\eta \geq 0$ are learned parameters, and $I(u)$ is the vector of RGB values for pixel u in the image. Shotton et al. (2006) learn the parameters for the node and edge potentials using a shared boosting method. Each object segmentation instance has 68,160 nodes (the images are `213 x 320`) and either $k = 5$ or $k = 8$ labels. As noted in Kappes et al. (2015), the MRFs used in practice increasingly use potential functions that are learned from data, rather than set by hand. In our experimental results, we found that both the objective gap and Hamming distance bounds were very good for these instances (in comparison to the stereo examples, which have "hand-set" potentials). Do the learning dynamics automatically encourage solutions to perturbed instances to be close to solutions of the original instance? Understanding the relationship between learning and this "stability" property is an interesting direction for future work.

C.2. Stereo Vision

In these models, the weights w_{uv} and costs $\theta_u(i)$ are set "by hand" according to the model from Tappen & Freeman (2003). Given two images taken from slightly offset locations, the goal is to estimate the depth of every pixel in one of the images. This can be done by estimating, for each pixel, the disparity between the two images, since the depth is inversely proportional to the disparity. In the Tappen & Freeman (2003) model, the node costs are set using the sampling-invariant technique from Birchfield & Tomasi (1998). These costs are similar to

$$\theta_u(i) = (I_L(u) - I_R(u - i))^2,$$

where I_L and I_R are the pixel intensities in the left and right images. If node u corresponds to pixel location (h, w), we use $u - i$ to represent the pixel in location $(h, w - i)$. So this cost function measures how likely it is that the pixel at location u in the left image corresponds to the pixel at location $u - i$ in the right image. The Birchfield-Tomasi matching costs are set using a correction to this expression that accounts for image sampling. In the Tappen and Freeman model, the weights w_{uv} are set as:

$$w_{uv} = \begin{cases} P \times s & |I(u) - I(v)| < T \\ s & \text{otherwise,} \end{cases}$$

where P, T, and s are the parameters of the model, and $I(u)$ is the intensity of pixel u in one of the input images to the stereo problem (in our experiments, we use I_L, the left image). These edge weights charge more for separating pixels with similar intensities, since nearby pixels with similar intensities are likely to correspond to the same object, and therefore be at the same depth. In our experiments, we follow Tappen & Freeman (2003) and set $s = 50$, $P = 2$, $T = 4$. In our experiments,

[1]All OpenGM 2 benchmark models are accessible at `http://hciweb2.iwr.uni-heidelberg.de/opengm/index.php?l0=benchmark`

we used images from the Middlebury stereo dataset (see, e.g., Scharstein et al., 2014). We used a downscaled version of the `tsukuba` image that was `120 x 150`, and had $k = 7$. Our `venus` model used the full-size image, which is `383 x 434`, and has $k = 5$. For `plastic`, we again used a downscaled, `111 x 127` image with $k = 5$. The large size of the `venus` image, in particular, shows that our verification algorithm is tractable to run even on fairly large problems.