

---

# Beyond Perturbation Stability: LP Recovery Guarantees for MAP Inference on Noisy Stable Instances

---

Hunter Lang\*  
MIT  
hjl@mit.edu

Aravind Reddy\*  
Northwestern University  
arareddy@u.northwestern.edu

David Sontag  
MIT  
dsontag@mit.edu

Aravindan Vijayaraghavan  
Northwestern University  
aravindv@northwestern.edu

## Abstract

Several works have shown that *perturbation stable* instances of the MAP inference problem in Potts models can be solved exactly using a natural linear programming (LP) relaxation. However, most of these works give few (or no) guarantees for the LP solutions on instances that do not satisfy the relatively strict perturbation stability definitions. In this work, we go beyond these stability results by showing that the LP approximately recovers the MAP solution of a stable instance even after the instance is corrupted by noise. This “noisy stable” model realistically fits with practical MAP inference problems: we design an algorithm for finding “close” stable instances, and show that several real-world instances from computer vision have nearby instances that are perturbation stable. These results suggest a new theoretical explanation for the excellent performance of this LP relaxation in practice.

## 1 Introduction

In this work, we study the MAP inference problem in the *ferromagnetic Potts model*, which is also known as uniform metric labeling (Kleinberg & Tardos, 2002). Given a graph  $G = (V, E)$ , this problem is:

$$\text{minimize}_{x: V \rightarrow [k]} \sum_{u \in V} c(u, x(u)) + \sum_{(u,v) \in E} w(u,v) \mathbb{1}[x(u) \neq x(v)].$$

Here we are optimizing over *labelings*  $x : V \rightarrow [k]$  where  $[k] = \{1, 2, \dots, k\}$ . The objective is comprised of “node costs”  $c : V \times [k] \rightarrow \mathbb{R}$ , and “edge weights”

$w : E \rightarrow \mathbb{R}_{>0}$ ; a labeling  $x$  pays the cost  $c(u, i)$  when it labels node  $u$  with label  $i$  and pays  $w(u, v)$  on edge  $(u, v)$  when it labels  $u$  and  $v$  differently. This problem is NP-hard for variable  $k \geq 3$  (Kleinberg & Tardos, 2002) even when the graph  $G$  is planar (Dahlhaus et al., 1992). However, there are several efficient and empirically successful approximation algorithms for the MAP inference problem—such as TRW (Wainwright et al., 2005) and MPLP (Globerson & Jaakkola, 2008)—that are related in some way to the *local LP relaxation*, which is also sometimes called the *pairwise LP* (Wainwright & Jordan, 2008; Chekuri et al., 2001). This LP relaxation returns an approximate MAP solution for most problem instances. However, when the parameters of these models are learned so as to enable good structured prediction, often the LP relaxation exactly or almost exactly recovers the MAP solution (Meshi et al., 2019). The connection between the LP relaxation and commonly used approximate MAP inference algorithms then leads to the following compelling question, which is of great practical relevance for understanding the “tightness” of the LP solution (informally, how close the LP solution is to the MAP solution).

*Can we explain the exceptional performance of the local LP relaxation in recovering the MAP solution in practice?*

Several works have studied different conditions that imply the local relaxation or related relaxations are tight (e.g., Kolmogorov & Wainwright, 2005; Wainwright & Jordan, 2008; Thapper et al., 2012; Weller et al., 2016; Rowland et al., 2017). Recent work on tightness of the local relaxation has focused on a class of several related conditions known as *perturbation stability*. Intuitively, an instance is perturbation stable if the solution  $x^*$  to the MAP inference problem is unique, and moreover,  $x^*$  is the unique solution even when the edge weights  $w$  are multiplicatively perturbed by a certain adversarial amount (Bilu & Linial, 2010). This structural assumption about the instance  $(G, c, w)$  captures the intuition that, on “real-world” instances, the ground-truth solution is stable and does not change much when the

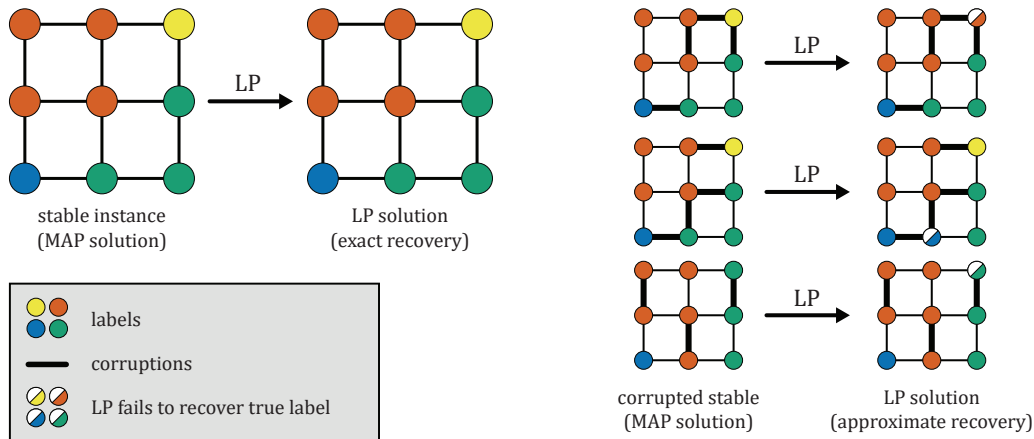


Figure 1: Left: prior work (Lang et al., 2018) showed that a *stable instance* can be exactly solved efficiently. Colors indicate the label of each vertex in the MAP solution  $x^*$ . On stable instances, solving the LP relaxation (represented by the arrow) recovers the MAP solution. However, real-world instances are *not* suitably stable for this result to apply in practice (Lang et al., 2019). Right: in this work, we show that solving the LP relaxation on a (slightly) *corrupted* stable instance (corruptions shown as bold edges) *approximately* recovers the original MAP solution. This is true even if the corruption changes the MAP solution (as in the bottom example). In other words, we prove that “easy” instances are still approximately easy even after small perturbations.

weights are slightly perturbed.

For constants  $\beta, \gamma \geq 1$ , we say that  $w'$  is a  $(\beta, \gamma)$ -perturbation of the weights  $w$  if  $\frac{1}{\beta} \cdot w(u, v) \leq w'(u, v) \leq \gamma \cdot w(u, v)$  for all  $(u, v) \in E$ . Suppose  $x^*$  is the unique MAP solution to the instance  $(G, c, w)$ . Then, we say  $(G, c, w)$  is a  $(\beta, \gamma)$ -stable instance if  $x^*$  is also the unique MAP solution to every instance  $(G, c, w')$  where  $w'$  is a  $(\beta, \gamma)$ -perturbation of  $w$ . Lang et al. (2018) showed that when  $(G, c, w)$  is  $(2, 1)$ -stable, the solution to the local LP relaxation is *persistent* i.e., the LP solution exactly recovers the MAP solution  $x^*$ .

While theoretically interesting,  $(2, 1)$ -stability is a strict condition that is unlikely to be satisfied in practice: the solution  $x^*$  is not allowed to change *at all* when the weights are perturbed. No real-world instances have yet been shown to be  $(2, 1)$ -stable (Lang et al., 2019). Moreover, the LP relaxation is also not persistent on most of those instances. However, the solution of the local LP relaxation is still *nearly persistent* i.e., the LP solution is very close to the MAP solution  $x^*$  (see Definition 3.1 for a formal definition). Those examples made it clear that theory must go beyond perturbation stability to explain this phenomenon of near-persistence that is prevalent in practice (see e.g., Sontag, 2010; Shekhovtsov et al., 2017).

*Why is the LP relaxation nearly persistent on MAP inference instances in practice?*

There are several theoretical frameworks to explain exactness or tightness of LP relaxations, such as total unimodularity, submodularity (Kolmogorov & Wain-

wright, 2005), and perturbation stability (Lang et al., 2018, 2019), as well as structural assumptions about the graph  $G$  (Wainwright & Jordan, 2008), or combined assumptions about  $G$  and the form of the objective function (Weller et al., 2016; Rowland et al., 2017). However, these frameworks can not be used to prove near-persistence.

Figure 1 (informally) shows our main result. The left side depicts the previous result of Lang et al. (2018): if the instance is  $(2, 1)$ -stable (a fairly strong structural assumption), the LP relaxation exactly recovers the full solution  $x^*$ . This result is limited because real-world instances have been shown to not satisfy  $(2, 1)$ -stability (Lang et al., 2019). The right side shows our main result: if the instance is a *slightly corrupted*  $(2, 1)$ -stable instance, the LP relaxation still *approximately* recovers the solution  $x^*$  to the stable instance.

Intuitively, we may expect a real-world instance to be “close” to a stable instance (i.e., to be a “corrupted stable” instance, as in Figure 1) even if the instance itself is not stable. We design an algorithm to check whether this is the case. We find that on several real examples, sparse and small-norm changes to the instance make it appropriately stable for our theorems to apply. In other words, we certify that these real instances are close to stable instances. For these instances, our theoretical results explain why the LP relaxation approximately recovers the MAP solution.

More formally, we assume that there is some *latent* stable instance  $(G, \bar{c}, \bar{w})$ , and that the observed instance

$(G, \hat{c}, \hat{w})$  is a noisy version of  $(G, \bar{c}, \bar{w})$  that is close to it. Let  $\hat{x}$  be the solution to the local LP on the observed instance  $(G, \hat{c}, \hat{w})$ , and let  $\bar{x}$  be the (unknown) MAP solution on the unseen stable instance  $(G, \bar{c}, \bar{w})$ . We prove that under certain conditions, the LP solution  $\hat{x}$  is *nearly persistent* i.e., the Hamming error  $\|\hat{x} - \bar{x}\|_1$  is small (see Definition 3.1). In other words, the local LP solution to the observed instance approximately recovers the latent integral solution  $\bar{x}$ .

We complement this by studying a natural generative model that generates noisy stable instances which, with high probability, satisfy the above conditions for *near persistency*. In other words, the observed instance  $(G, \hat{c}, \hat{w})$  is obtained by random perturbations to the latent stable instance  $(G, \bar{c}, \bar{w})$ , and the LP relaxation approximately recovers the MAP solution to the latent instance with high probability.

Our theoretical results imply that the local LP approximately recovers the MAP solution when the observed instance is close to a stable instance. Our empirical results suggest that real-world instances are very close to stable instances. These results together suggest a new explanation for the near-persistency of the solution of the local LP relaxation for MAP inference in practice. To prove these results and derive our algorithm for finding a “close-by” stable instance, we make several novel technical contributions, which we outline below.

### Technical contributions.

- In Section 4, we generalize the  $(2, 1)$ -stability result of Lang et al. (2018) to work under a much weaker assumption, which we call  $(2, 1)$ -expansion stability. That is, we prove the local LP is tight on  $(2, 1)$ -expansion stable instances. Additionally, given the instance’s MAP solution,  $(2, 1)$ -expansion stability is efficiently checkable. To the best of our knowledge, most other perturbation stability assumptions are not known to satisfy this desirable property. This generalization is crucial for the efficiency of our algorithm for finding stable instances that are close to a given observed instance.
- In Section 5, we give a simple extension of  $(2, 1)$ -expansion stability called  $(2, 1, \psi)$ -expansion stability. We prove it implies a “curvature” result around the MAP solution  $\bar{x}$ . On instances that satisfy this condition, if a labeling  $\hat{x}$  is close in objective value to  $\bar{x}$ , it must also be close in the solution space. This result lets us translate between objective gap and Hamming distance.
- In Section 6, we study a natural generative model where the observed instance is generated from an arbitrary *latent stable* instance by random (sub-Gaussian) perturbations to the costs and weights.

We prove that, with high probability, every feasible LP solution takes close objective values on the latent and observed instances. The proof uses a rounding algorithm for metric labeling in a novel way to obtain stronger guarantees. When combined with our other results, this proves that when the latent instance is  $(2, 1, \psi)$ -expansion stable, the LP solution is nearly persistent on the observed instance with high probability. These results suggest a theoretical explanation for the phenomenon of near-persistence of the LP solution in practice.

- We design an efficient algorithm for finding  $(2, 1, \psi)$ -expansion stable instances that are “close” to a given instance  $(G, \hat{c}, \hat{w})$  in Section 7. To the best of our knowledge, this is the first algorithm for finding close-by stable instances, and is also an efficient algorithm for checking  $(2, 1, \psi)$ -expansion stability. This algorithm allows us to check whether real-world instances can plausibly be considered “corrupted stable” instances as shown in Figure 1.
- We run our algorithm on several real-world instances of MAP inference in Section 8, and find that the observed instances  $(G, \hat{c}, \hat{w})$  often admit close-by  $(2, 1, \psi)$ -stable instances  $(G, \bar{c}, \bar{w})$ . Moreover, we find that the local LP solution  $\hat{x}$  typically has very close objective to  $\bar{x}$  in  $(G, \bar{c}, \bar{w})$ . Our curvature result for  $(2, 1, \psi)$ -stable instances thus gives an explanation for the tightness of the local LP relaxation on  $(G, \hat{c}, \hat{w})$ .

## 2 Related work

**Perturbation stability.** Several works have given recovery guarantees for the local LP relaxation on perturbation stable instances of uniform metric labeling (Lang et al., 2018, 2019) and for similar problems (Makarychev et al., 2014; Angelidakis et al., 2017). Lang et al. (2019) give partial recovery guarantees for the local LP when parts (*blocks*) of the observed instance satisfy a stability-like condition, and they showed that practical instances have blocks that satisfy their condition. However, the required *block stability* condition in turn depends on certain quantities related to the LP dual. This is unsatisfactory since this does not explain when and why such instances are likely to arise in practice. For a more extensive treatment of the subject, we refer the reader to the “Perturbation Resilience” chapter from Roughgarden (2021).

**Easy instances corrupted with noise.** Our random noise model is similar to several planted average-case models like stochastic block models (SBMs) considered in the context of problems like community detection, correlation clustering and partitioning (see e.g.,

McSherry, 2001; Abbe, 2018; Globerson et al., 2015). Instances generated from these models can also be seen as the result of random noise injected into an instance with a nice block structure that is easy to solve. Several works give exact recovery and approximate recovery guarantees for semidefinite programming (SDP) relaxations for such models in different parameter regimes (Abbe, 2018; Guédon & Vershynin, 2016). In our model however, we start with an *arbitrary* stable instance as opposed to an instance with a block structure (which is trivial to solve). Moreover, we are unaware of such analysis in the context of linear programs. Please see Section 6 for a more detailed comparison. To the best of our knowledge, we are the first to study instances generated from random perturbations to stable instances.

**Partial optimality algorithms.** Several works have developed fast algorithms for identifying parts of the MAP assignment. These algorithms output an approximate solution  $\hat{x}$  and a set of vertices where  $\hat{x}$  provably agrees with the MAP solution  $x^*$  (e.g., Kovtun, 2003; Shekhovtsov, 2013; Swoboda et al., 2016; Shekhovtsov et al., 2017). Like these works, our results also prove that an approximate solution  $\hat{x}$  has small error  $|\hat{x} - x^*|$ . However, these previous approaches are more concerned with designing fast algorithms for finding such  $\hat{x}$ . In contrast, we focus on giving structural conditions that explain why a *particular*  $\hat{x}$  (the solution to the local LP relaxation) often approximately recovers  $x^*$ . Our algorithm in Section 7 is thus not meant as an efficient method for certifying that  $|\hat{x} - x^*|$  is small, but rather as a method for checking whether our structural condition (that the observed instance is close to a stable instance) is satisfied in practice.

### 3 Preliminaries

In this section we introduce our notation, define the local LP relaxation for MAP inference, and give more details on perturbation stability. As in the previous section, the MAP inference problem in the ferromagnetic Potts model on the instance  $(G, c, w)$  can be written in *energy minimization* form as:

$$\underset{x: V \rightarrow [k]}{\text{minimize}} \sum_{u \in V} c(u, x(u)) + \sum_{(u, v) \in E} w(u, v) \mathbb{1}[x(u) \neq x(v)]. \quad (1)$$

Here  $x$  is an *assignment* (or labeling) of vertices to labels i.e.  $x: V \rightarrow \{1, 2, \dots, k\}$ . We can identify each labeling  $x$  with a point  $(x_u: u \in V; x_{uv}: (u, v) \in E)$ , where each  $x_u \in \{0, 1\}^k$  and each  $x_{uv} \in \{0, 1\}^{k \times k}$ .

In this work, we consider all node costs  $c(u, i) \in \mathbb{R}$  and all edge weights  $w(u, v) > 0$ . We note that this is equivalent to the formulation where all node costs and

edge weights are non-negative (Kleinberg & Tardos, 2002). See Appendix A for a proof of this equivalence.

We encode the node costs and the edge weights in a vector  $\theta \in \mathbb{R}^{nk+mk^2}$  where  $n = |V|$  and  $m = |E|$  s.t.  $\theta(u, i) = c(u, i)$ ,  $\theta(u, v, i, j) = w(u, v) \mathbb{1}[i \neq j]$ . Then the objective can be written as  $\langle \theta, x \rangle$ . We set  $x_u(i) = 1$  when  $x(u) = i$ , and 0 otherwise. Similarly, we set  $x_{uv}(i, j) = 1$  when  $x(u) = i$  and  $x(v) = j$ , and 0 otherwise. Where convenient, we use  $x$  to refer to this point rather than the labeling  $x: V \rightarrow [k]$ . We can then rewrite (1) as:

$$\begin{aligned} \min_x \quad & \sum_{u \in V} \sum_{i=1}^k c(u, i) x_u(i) + \sum_{(u, v) \in E} w(u, v) \sum_{i \neq j} x_{uv}(i, j) \\ \text{subject to:} \quad & \sum_{i=1}^k x_u(i) = 1 \quad \forall u \in V \\ & \sum_{i=1}^k x_{uv}(i, j) = x_v(j) \quad \forall (u, v) \in E, j \in [k] \\ & \sum_{j=1}^k x_{uv}(i, j) = x_u(i) \quad \forall (u, v) \in E, i \in [k] \\ & x_{uv}(i, j) \in \{0, 1\} \quad \forall (u, v), (i, j) \\ & x_u(i) \in \{0, 1\} \quad \forall u, i. \end{aligned}$$

This is equivalent to (1), and is an integer linear program (ILP). By relaxing the integrality constraints from  $\{0, 1\}$  to  $[0, 1]$ , we obtain the *local LP relaxation*:

$$\min_{x \in L(G)} \sum_{u \in V} \sum_{i=1}^k c(u, i) x_u(i) + \sum_{(u, v) \in E} w(u, v) \sum_{i \neq j} x_{uv}(i, j),$$

where  $L(G)$  is the polytope defined by the same constraints as above, with  $x \in \{0, 1\}$  replaced with  $x \in [0, 1]$ . This is known as the *local polytope* (Wainwright & Jordan, 2008). The vertices of  $L(G)$  are either *integral*, meaning all  $x_u$  and  $x_{uv}$  take values in  $\{0, 1\}$ , or *fractional*, when some variables take values in  $(0, 1)$ . Integral vertices of this polytope correspond to labelings  $x: V \rightarrow [k]$ , so if the LP solution is obtained at an integral vertex, then it is also a MAP assignment.

If the solution  $x^*$  of this relaxation on an instance  $(G, c, w)$  is obtained at an integral vertex, we say the LP is *tight* on the instance, because the LP has exactly recovered a MAP assignment. If the LP is not tight, there may still be some vertices  $u$  where  $x_u^*$  takes integral values. In this case, if  $x_u^*(i) = 1$  and  $\bar{x}(u) = i$ , i.e., the LP solution agrees with the MAP assignment  $\bar{x}$  at vertex  $u$ , the LP is said to be *persistent* at  $u$ .  $x_u^*(i) \in \{0, 1\}$  does not imply the LP is persistent at  $u$ , in general. The LP solution  $x^*$  is said to be persistent if it agrees with  $\bar{x}$  at every vertex  $u \in V$ .



*Recovery error:* In practice, the local LP relaxation is often not tight, but is *nearly persistent*. We will measure the recovery error of our LP solution in terms of the ‘‘Hamming error’’ between the LP solution and the MAP assignment.

**Definition 3.1** (Recovery error). Given an instance  $(G, c, w)$  of (1), let  $\bar{x}$  be a MAP assignment, and let  $x^*$  be a solution to the local LP relaxation. The recovery error is given by (with some abuse of notation)

$$\begin{aligned} \frac{1}{2} \|x^* - \bar{x}\|_1 &:= \frac{1}{2} \|x_V^* - \bar{x}_V\|_1 \\ &= \frac{1}{2} \sum_{u \in V} \sum_{i \in [k]} |x_u^*(i) - \mathbb{1}[\bar{x}(u) = i]|. \end{aligned}$$

$x_V \in \mathbb{R}^{nk}$  denotes the portion of  $x$  restricted to the vertex set  $V$ . If  $x^*$  is integral, the recovery error measures the number of vertices where  $x^*$  disagrees with  $\bar{x}$ . When the recovery error of  $x^*$  is 0, the solution  $x^*$  is *persistent*. We will say that the LP solution  $x^*$  is *nearly persistent* when the recovery error of solution  $x^*$  is a small fraction of  $n$ .

In our analysis, we will consider the following subset  $L^*(G)$  of  $L(G)$  which is easier to work with, and which contains all points we are interested in.

**Definition 3.2** ( $L^*(G)$ ). We define  $L^*(G) \subseteq L(G)$  to be the set of points  $x \in L(G)$  which further satisfy the constraint that  $x_{uv}(i, i) = \min(x_u(i), x_v(i))$  for all  $(u, v) \in E$  and  $i \in [k]$ .

**Claim 3.3.** *For a given graph  $G$ , every solution  $x \in L(G)$  that minimizes  $\langle \theta, x \rangle$  for some valid objective vector  $\theta = (c, w)$  also belongs to  $L^*(G)$ . Further, all integer solutions in  $L(G)$  also belong to  $L^*(G)$ .*

We prove this claim in Appendix A.

Our new stability result relies on the set of *expansions* of a labeling  $x$ .

**Definition 3.4** (Expansion). Let  $x : V \rightarrow [k]$  be a labeling of  $V$ . For any label  $\alpha \in [k]$ , we say that  $x'$  is an  $\alpha$ -expansion of  $x$  if  $x' \neq x$  and the following hold for all  $u \in V$ :

$$\begin{aligned} x(u) = \alpha &\implies x'(u) = \alpha, \\ x'(u) \neq \alpha &\implies x'(u) = x(u). \end{aligned}$$

That is,  $x'$  may only expand the set of points labeled  $\alpha$ , and cannot make other changes to  $x$ .

## 4 Expansion Stability

In this section, we generalize the stability result of Lang et al. (2018) to a much broader class of instances. This generalization allows us to efficiently check whether a

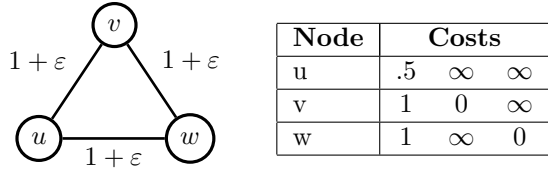


Figure 2:  $(2, 1)$ -expansion stable instance that is not  $(2, 1)$ -stable. In the original instance (shown left), the optimal solution labels each vertex with label 1, for an objective of 2.5. This instance is not  $(2, 1)$ -stable: consider the  $(2, 1)$ -perturbation that multiplies all edge weights by  $1/2$ . In this perturbed instance, the original solution still has objective 2.5, and the new optimal solution labels  $(u, v, w) \rightarrow (1, 2, 3)$ . This has a node cost of 0.5 and an edge cost of  $(3 + 3\epsilon)/2$ , for a total of  $2 + 3\epsilon/2 < 2.5$ . Since the original solution is not optimal in the perturbed instance, this instance is not  $(2, 1)$ -perturbation stable. However, note that the only expansions of the original solution (which had all label 1) that have non-infinite objective are  $(u, v, w) \rightarrow (1, 2, 1)$  and  $(u, v, w) \rightarrow (1, 1, 3)$ . These each have objective  $2.5 + \epsilon$ , which is strictly greater than the perturbed objective of the original solution. In fact, checking this single perturbation, known as the *adversarial perturbation* is enough to verify expansion stability: this instance is  $(2, 1)$ -expansion stable. We include the full details in Appendix B.

real-world instance could plausibly have the structure shown in Figure 1 (that is, whether the instance is close to a suitably stable instance).

Consider a fixed instance  $(G, c, w)$  with a unique MAP solution  $\bar{x}$ . Theorem 1 of Lang et al. (2018) requires that for all  $\theta' \in \{(c, w') \mid w' \in \{(2, 1)\text{-perturbations of } w\}\}$ ,  $\langle \theta', x \rangle > \langle \theta', \bar{x} \rangle$  for all labelings  $x \neq \bar{x}$ . That is, that result requires  $\bar{x}$  to be the unique optimal solution in any  $(2, 1)$ -perturbation of the instance. By contrast, our result only requires  $\bar{x}$  to have better perturbed objective than the set of *expansions* of  $\bar{x}$  (c.f. Definition 3.4).

**Definition 4.1** ( $(2, 1)$ -expansion stability). Let  $\bar{x}$  be the unique MAP solution for  $(G, c, w)$ , and let  $\mathcal{E}_{\bar{x}}$  be the set of expansions of  $\bar{x}$  (see Definition 3.4). Let

$$\Theta = \{(c, w') \mid w' \in \{(2, 1)\text{-perturbations of } w\}\}$$

be the set of all objective vectors within a  $(2, 1)$ -perturbation of  $\theta = (c, w)$ . We say the instance  $(G, c, w)$  is  $(2, 1)$ -expansion stable if the following holds for all  $\theta' \in \Theta$  and all  $x \in \mathcal{E}_{\bar{x}}$ :

$$\langle \theta', x \rangle > \langle \theta', \bar{x} \rangle.$$

That is,  $\bar{x}$  is better than all of its expansions  $x \neq \bar{x}$  in every  $(2, 1)$ -perturbation of the instance.

**Theorem 4.2** (Local LP is tight on  $(2, 1)$ -expansion stable instances). *Let  $\bar{x}$  and  $\hat{x}$  be the MAP and local LP solutions to a  $(2, 1)$ -expansion stable instance  $(G, c, w)$ , respectively. Then  $\bar{x} = \hat{x}$  i.e. the local LP is tight on  $(G, c, w)$ .*

We defer the proof of this theorem to Appendix B as it is similar to the proof of Theorem 1 from Lang et al. (2018). The  $(2, 1)$ -expansion stability assumption is much weaker than  $(2, 1)$ -stability because the former only compares  $\bar{x}$  to its expansions, whereas the latter compares  $\bar{x}$  to *all* labelings. While the rest of our results can also be adapted to the  $(2, 1)$ -stability definition, this relaxed assumption gives better empirical results. Figure 2 shows an example of a  $(2, 1)$ -expansion stable instance that is not  $(2, 1)$ -stable. This shows that our new stability condition is less restrictive.

## 5 Curvature around MAP solution and near persistence of the LP solution

In this section, we show that a condition related to  $(2, 1)$ -expansion stability, called  $(2, 1, \psi)$ -expansion stability, implies a “curvature” result for the objective function around the MAP solution  $\bar{x}$ . On instances satisfying this condition, any point  $\hat{x} \in L(G)$  with objective close to  $\bar{x}$  also has small  $\|\hat{x} - \bar{x}\|_1$ , so  $\hat{x}$  and  $\bar{x}$  are close in solution space. In other words, if the LP solution  $\hat{x}$  to a “corrupted”  $(2, 1, \psi)$ -expansion stable instance is near-optimal in the original  $(2, 1, \psi)$ -expansion stable instance (whose solution is  $\bar{x}$ ), then the result in this section implies  $\|\hat{x} - \bar{x}\|_1$  is small. This immediately gives a version of the result in the right panel of Figure 1: suppose we define an instance to be *close* to a  $(2, 1, \psi)$ -expansion stable  $(G, \bar{c}, \bar{w})$  if its LP solution  $\hat{x}$  is approximately optimal in  $(G, \bar{c}, \bar{w})$ . Then the curvature result implies that the LP approximately recovers the stable instance’s MAP solution  $\bar{x}$  for all close instances. In Section 6, we give a generative model where the generated instances are “close” according to this definition with high probability.

The  $(2, 1, \psi)$ -expansion stability condition, for  $\psi > 0$ , says that the instance is  $(2, 1)$ -expansion stable even if we allow all node costs  $c(u, i)$  to be additively perturbed by up to  $\psi$ . This extra additive stability will allow us to prove the curvature result. This is related to the use of additive stability in Lang et al. (2019) to give persistency guarantees.

**Definition 5.1** ( $(2, 1, \psi)$ -expansion stable). For  $\psi > 0$ , we say an instance  $(G, c, w)$  is  $(2, 1, \psi)$ -expansion stable if  $(G, c', w)$  is  $(2, 1)$ -expansion stable for all  $c'$  with  $c \leq c' \leq c + \psi \cdot \mathbf{1}$  where  $\mathbf{1}$  is the all-ones vector.

The following theorem shows low recovery error i.e.,

near persistence of the LP solution on  $(2, 1, \psi)$  expansion stable instances in terms of the gap in objective value.

**Theorem 5.2.** *Let  $(G, c, w)$  be a  $(2, 1, \psi)$ -expansion stable instance with MAP solution  $\bar{x}$ . Let  $\theta = (c, w)$ . Then for any  $x \in L^*(G)$ , the recovery error (see Def. 3.1) satisfies*

$$\frac{1}{2}\|x - \bar{x}\|_1 := \frac{1}{2}\|x_V - \bar{x}_V\|_1 \leq \frac{1}{\psi}|\langle \theta, x \rangle - \langle \theta, \bar{x} \rangle|. \quad (2)$$

*Proof (sketch).* For any  $x \in L^*(G)$ , we construct a feasible solution  $\hat{x}$  which is a strict convex combination of  $x$  and  $\bar{x}$  that is very close to  $\bar{x}$ . Then, we apply a rounding algorithm to  $\hat{x}$  to get a random integer solution  $h$ . Let  $\hat{\theta}$  represent the worst  $(2, 1)$ -perturbation for  $\bar{x}$ . This is the instance where all the edges not cut by  $\bar{x}$  have their weights multiplied by  $1/2$ . We define the objective difference using  $\hat{\theta}$  as  $A_h = \langle \hat{\theta}, h \rangle - \langle \hat{\theta}, \bar{x} \rangle$ . First we show an upper bound for  $\mathbb{E}[A_h]$  using properties of the rounding algorithm. Then we show that for any solution  $h$  in the support of our rounding algorithm,  $A_h \geq \psi \cdot B_h$  where  $B_h$  is the Hamming error of  $h$  (when compared to  $\bar{x}$ ). On the other hand, one can also use the properties of the rounding algorithm to get a lower bound on  $\mathbb{E}[B_h]$  in terms of the recovery error (i.e., Hamming error) of the LP solution. These bounds together imply the required upper bound on the recovery error of the LP solution.  $\square$

We defer the complete proof and an alternate dual-based proof to Appendix C.

Theorem 5.2 shows that on a  $(2, 1, \psi)$ -expansion stable instance, small objective gap  $\langle \theta, x \rangle - \langle \theta, \bar{x} \rangle$  implies small distance  $\|x_V - \bar{x}_V\|_1$  in solution space. Although this holds for any  $x \in L^*(G)$ , we will be interested in  $x$  that are LP solutions to an observed, corrupted version of the stable instance.

We now show that if the observed instance has a nearby stable instance, then the LP solution for the observed instance has small Hamming error. For any two instances  $\hat{\theta} = (\hat{c}, \hat{w})$  and  $\bar{\theta} = (\bar{c}, \bar{w})$  on the same graph  $G$ , the metric between them  $d(\hat{\theta}, \bar{\theta}) := \sup_{x \in L^*(G)} |\langle \hat{\theta}, x \rangle - \langle \bar{\theta}, x \rangle|$ .

**Corollary 5.3** (LP solution is good if there is a nearby stable instance). *Let  $\hat{x}^{MAP}$  and  $\hat{x}$  be the MAP and local LP solutions to an observed instance  $(G, \hat{c}, \hat{w})$ . Also, let  $\bar{x}$  be the MAP solution for a latent  $(2, 1, \psi)$ -expansion stable instance  $(G, \bar{c}, \bar{w})$ . If  $\hat{\theta} = (\hat{c}, \hat{w})$  and  $\bar{\theta} = (\bar{c}, \bar{w})$ ,*

$$\frac{1}{2}\|\hat{x}_V - \hat{x}_V^{MAP}\|_1 \leq \frac{2d(\hat{\theta}, \bar{\theta})}{\psi} + \frac{1}{2}\|\hat{x}_V^{MAP} - \bar{x}_V\|_1.$$

We defer the proof of this corollary to Appendix C.

## 6 Generative model for noisy stable instances

In the previous section, we showed that if an instance  $(G, \hat{c}, \hat{w})$  is “close” to a  $(2, 1, \psi)$ -expansion stable instance  $(G, \bar{c}, \bar{w})$  (i.e., the LP solution  $\hat{x}$  to  $(G, \hat{c}, \hat{w})$  has good objective in  $(G, \bar{c}, \bar{w})$ ), then  $\|\hat{x} - \bar{x}\|$  is small, where  $\bar{x}$  is the MAP solution to the stable instance. In this section, we give a natural generative model for  $(G, \hat{c}, \hat{w})$ , based on randomly corrupting  $(G, \bar{c}, \bar{w})$ , in which  $\hat{x}$  has good objective in  $(G, \bar{c}, \bar{w})$  with high probability. Together with the curvature result from the previous section (Theorem 5.2), this implies that the LP relaxation, run on the noisy instances  $(G, \hat{c}, \hat{w})$ , approximately recovers  $\bar{x}$  with high probability.

We now describe our generative model for the problem instances, which starts with an arbitrary stable instance and perturbs it with random additive perturbations to the edge costs and node costs (of potentially varying magnitudes). The random perturbations reflect possible uncertainty in the edge costs and node costs of the Markov random field. We will assume the random noise comes from any distribution that is sub-Gaussian<sup>1</sup>. However, there is a small technicality: the edge costs need to be positive (node costs can be negative). For this reason we will consider truncated sub-Gaussian random variables for the noise for the edge weights. We define sub-Gaussian and truncated sub-Gaussian random variables in Appendix D.

**Generative Model:** We start with an instance  $(G, \bar{c}, \bar{w})$  that is  $(2, 1, \psi)$ -expansion stable, and perturb the edge costs and node costs independently. Given any instance  $(G, \bar{c}, \bar{w})$ , an instance  $(G, \hat{c}, \hat{w})$  from the model is generated as follows:

1. For all node-label pairs  $(u, i)$ ,  $\hat{c}(u, i) = \bar{c}(u, i) + \tilde{c}(u, i)$ , where  $\tilde{c}(u, i)$  is sub-Gaussian with mean 0 and parameter  $\sigma_{u,i}$ .
2. For all edges  $(u, v)$ ,  $\hat{w}(u, v) = \bar{w}(u, v) + \tilde{w}(u, v)$ , where  $\tilde{w}(u, v)$  is an independent r.v. that is  $(-w(u, v), \gamma_{u,v})$ -truncated sub-Gaussian with mean 0.
3.  $(G, \hat{c}, \hat{w})$  is the observed instance.

By the definition of our model, the edge weights  $\hat{w}(u, v) \geq 0$  for all  $(u, v) \in E$ . The parameters of the model are the unperturbed instance  $(G, \bar{c}, \bar{w})$ , and the noise parameters  $\{\gamma_{u,v}, \sigma_{u,i}\}_{u,v \in V, i \in [k]}$ . On the one hand, the above model captures a natural average-case model for the problem. For a fixed ground-truth solution  $x^* : V \rightarrow [k]$ , consider the stable instance  $(H, c, w)$  where  $w_{uv}^* = 2$  for all  $u, v$  in the same clus-

ter (i.e.,  $x^*(u) = x^*(v)$ ) and  $w_{uv}^* = 1$  otherwise; and with  $c^*(u, i) = 1$  if  $x^*(u) = i$ , and  $c^*(u, i) = 1 + \psi$  otherwise. The above noisy stable model with stable instance  $(H, c, w)$  generates instances that are reminiscent of (stochastic) block models, with additional node costs. On the other hand, the above model is much more general, since we can start with *any* stable instance  $(G, c, w)$ .

With high probability over the random corruptions of our stable instance, the local LP on the corrupted instance approximately recovers the MAP solution  $\bar{x}$  of the stable instance. The key step in the proof of this theorem is showing that, with high probability, the observed instance is close to the latent stable instance in the metric we defined earlier.

**Lemma 6.1** ( $d(\hat{\theta}, \bar{\theta})$  is small w.h.p. ). *There exists a universal constant  $c < 1$  such that for any instance in the above model, with probability at least  $1 - o(1)$ ,*

$$\sup_{x \in L^*(G)} |\langle \hat{\theta}, x \rangle - \langle \bar{\theta}, x \rangle| \leq c\sqrt{nk} \sqrt{\sum_{u,i} \sigma_{u,i}^2 + \frac{k^2}{4} \sum_{uv} \gamma_{u,v}^2}$$

*Proof (sketch).* For any fixed  $x \in L^*(G)$ , we can show that  $|\langle \hat{\theta}, x \rangle - \langle \bar{\theta}, x \rangle|$  is small w.h.p. using a standard large deviations bound for sums of sub-Gaussian random variables. The main technical challenge is in showing that the supremum over all feasible solutions is small w.h.p. The standard approach is to perform a union bound over an  $\varepsilon$ -net of feasible LP solutions in  $L^*$ . However, this gives a loose bound. Instead, we upper bound the supremum by using a rounding algorithm for LP solutions in  $L^*(G)$ , and union bound only over the discrete solutions output by the rounding algorithm. This gives significant improvements over the standard approach; for example, in a  $d$ -regular graph with equal variance parameter  $\gamma_{uv}$ , this saves a factor of  $\sqrt{d}$  apart from logarithmic factors in  $n$ .  $\square$

We defer the details to Appendix D. The above proof technique that uses a rounding algorithm to provide a deviation bound for a continuous relaxation is similar to the analysis of SDP relaxations for average-case problems (see e.g., Makarychev et al., 2013; Guédon & Vershynin, 2016). The above lemma, when combined with Theorem 5.2 gives the following guarantee.

**Theorem 6.2** (LP solution is nearly persistent). *Let  $\hat{x}$  be the local LP solution to the observed instance  $(G, \hat{c}, \hat{w})$  and  $\bar{x}$  be the MAP solution to the latent  $(2, 1, \psi)$ -expansion stable instance  $(G, \bar{c}, \bar{w})$ . With high probability over the random noise,*

$$\frac{1}{2} \|\hat{x}_V - \bar{x}_V\|_1 \leq \frac{2}{\psi} \cdot c\sqrt{nk} \cdot \sqrt{\sum_{u,i} \sigma_{u,i}^2 + k^2 \sum_{uv} \gamma_{u,v}^2}$$

<sup>1</sup>All of the results that follow can also be generalized to sub-exponential random variables; however for convenience, we restrict our attention to sub-Gaussians.

*Proof.* We know that for any feasible solution  $x \in L(G)$ ,  $\langle \bar{\theta}, x \rangle \geq \langle \bar{\theta}, \bar{x} \rangle$ . Therefore,  $\langle \bar{\theta}, \hat{x} \rangle \geq \langle \bar{\theta}, \bar{x} \rangle$ . Remember that we defined  $d(\hat{\theta}, \bar{\theta})$  as  $\sup_{x \in L^*(G)} |\langle \hat{\theta}, x \rangle - \langle \bar{\theta}, x \rangle|$ . Since  $\hat{x}$  and  $\bar{x}$  are both points in  $L^*(G)$ ,

$$\begin{aligned} \langle \bar{\theta}, \hat{x} \rangle &\leq \langle \hat{\theta}, \hat{x} \rangle + d(\hat{\theta}, \bar{\theta}) \leq \langle \hat{\theta}, \bar{x} \rangle + d(\hat{\theta}, \bar{\theta}) \\ &\leq \langle \bar{\theta}, \bar{x} \rangle + 2d(\hat{\theta}, \bar{\theta}) \end{aligned}$$

The first and third inequalities follow from the definition of  $d(\hat{\theta}, \bar{\theta})$ . The second inequality follows from the fact that  $\hat{x}$  is the minimizer of  $\langle \hat{\theta}, x \rangle$  over all  $x \in L(G)$ . Therefore,  $0 \leq \langle \hat{\theta}, \hat{x} \rangle - \langle \bar{\theta}, \bar{x} \rangle \leq 2d(\hat{\theta}, \bar{\theta})$ . Using this in Theorem 5.2, we get  $\frac{1}{2} \|\hat{x} - \bar{x}\|_1 \leq \frac{2d(\hat{\theta}, \bar{\theta})}{\psi}$ . Lemma 6.1 then gives an upper bound on  $d(\hat{\theta}, \bar{\theta})$  that holds w.h.p.  $\square$

For a  $d$ -regular graph in the uniform setting, we get the following useful corollary:

**Corollary 6.3** (MAP solution recovery for regular graphs). *Suppose we have a  $d$ -regular graph  $G$  with  $\gamma_{u,v}^2 = \gamma^2$  for all edges  $(u, v)$ , and  $\sigma_{u,i}^2 = \sigma^2$  for all vertex-label pairs  $(u, i)$ . Also, suppose only a fraction  $\rho$  of the vertices and  $\eta$  of the edges are subject to the noise. With high probability over the random noise,*

$$\frac{\|\hat{x}_V - \bar{x}_V\|_1}{2n} \leq \frac{2ck\sqrt{\rho\sigma^2 + \frac{\eta dk}{8}\gamma^2}}{\psi}$$

Note that when  $\hat{x}$  is an integer solution, the left-hand-side is the fraction of vertices misclassified by  $\hat{x}$ .

## 7 Finding nearby stable instances

In this section, we describe an algorithm for efficiently finding  $(2, 1, \psi)$ -expansion stable instances that are “close” to an observed instance  $(G, \hat{c}, \hat{w})$ . This algorithm allows us to check whether we can plausibly model real-world instances as “corrupted” versions of a  $(2, 1, \psi)$ -expansion stable instance.

In addition to an observed instance  $(G, \hat{c}, \hat{w})$ , the algorithm takes as input a “target” labeling  $x^t$ . For example,  $x^t$  could be a MAP solution of the observed instance. Surprisingly, once given a target solution, this algorithm is efficient.

We want to search over costs  $c$  and weights  $w$ . The broad goal to solve the following optimization problem:

$$\begin{aligned} &\underset{c \geq 0, w \geq 0}{\text{minimize}} && f(c, w) && (3) \\ &\text{subject to} && (G, c, w) \text{ is } (2, 1, \psi)\text{-expansion stable} \\ & && \text{with MAP solution } x^t, \end{aligned}$$

where  $f(c, w)$  is any convex function of  $c$  and  $w$ . In particular, we will use  $f_1(c, w) = \|(c, w) - (\hat{c}, \hat{w})\|_1$  and

$f_2(c, w) = \frac{1}{2} \|(c, w) - (\hat{c}, \hat{w})\|_2^2$  for minimizing the L1 and L2 distances between to the observed instance. The output of this optimization problem will give the closest objective vector  $(\bar{c}, \bar{w})$  for which the instance  $(G, \bar{c}, \bar{w})$  is  $(2, 1, \psi)$ -expansion stable. If the optimal objective value of this optimization is 0, the observed instance  $(G, \hat{c}, \hat{w})$  is  $(2, 1, \psi)$ -expansion stable.

There is always a feasible  $(c, w)$  for (3) (see Appendix E for a proof), but it may change many weights and costs. Next we derive an efficiently-solvable reformulation of (3). In the next section, we find that the changes to  $\hat{c}$  and  $\hat{w}$  required to find a  $(2, 1, \psi)$ -expansion stable instance are relatively sparse in practice.

**Theorem 7.1.** *The optimization problem (3) can be expressed as a convex minimization problem over a polytope described by  $\text{poly}(n, m, k)$  constraints. When  $f(c, w) = \|(c, w) - (\hat{c}, \hat{w})\|_1$ , (3) can be expressed as a linear program.*

The instance  $(G, c, w)$  is  $(2, 1, \psi)$ -expansion stable if  $x^t$  is better than every expansion  $y$  of  $x^t$  in every  $(2, 1, \psi)$ -perturbation of  $(c, w)$ . Let  $\mathcal{E}$  be the set of all expansions of the target solution  $x^t$ . Then for all  $\theta'$  within a  $(2, 1, \psi)$ -perturbation of  $\theta = (c, w)$ , we should have that  $\langle \theta', x^t \rangle \leq \min_{y \in \mathcal{E}} \langle \theta', y \rangle$ . It is enough to check the adversarial  $(2, 1, \psi)$ -perturbation  $\theta_{adv}$ . This perturbation makes the target solution  $x^t$  as bad as possible. If  $x^t$  is better than all the expansions  $y \in \mathcal{E}$  in this perturbation, it is better than all  $y \in \mathcal{E}$  in every  $(2, 1, \psi)$ -perturbation (see Appendix E for a proof). We set  $\theta_{adv} = (c_{adv}, w_{adv})$  as:

$$\begin{aligned} c_{adv}(u, i) &= \begin{cases} c(u, i) + \psi & i = x^t(u), \\ c(u, i) & \text{otherwise.} \end{cases} \\ w_{adv}(u, v) &= \begin{cases} w(u, v) & x^t(u) \neq x^t(v), \\ \frac{1}{2}w(u, v) & \text{otherwise.} \end{cases} \end{aligned}$$

The target solution  $x^t$  is fixed, so  $\langle \theta_{adv}, x^t \rangle$  is a linear function of the optimization variables  $c$  and  $w$ . For  $\alpha \in [k]$ , let  $\mathcal{E}^\alpha$  be the set of  $\alpha$ -expansions of  $x^t$ . Because  $\mathcal{E} = \cup_{\alpha \in [k]} \mathcal{E}^\alpha$ , we have  $\langle \theta', x^t \rangle \leq \min_{y \in \mathcal{E}} \langle \theta', y \rangle$  if and only if  $\langle \theta', x^t \rangle \leq \min_{y \in \mathcal{E}^\alpha} \langle \theta', y \rangle$  for all  $\alpha \in [k]$ . We can simplify the original optimization problem to:

$$\begin{aligned} &\underset{c \geq 0, w \geq 0}{\text{minimize}} && f(c, w) \\ &\text{subject to} && \langle \theta_{adv}, x^t \rangle \leq \min_{y \in \mathcal{E}^\alpha} \langle \theta_{adv}, y \rangle \quad \forall \alpha \in [k], \end{aligned}$$

$\theta_{adv}$  is a linear function of  $c, w$  as defined above. We now use the structure of the sets  $\mathcal{E}^\alpha$  to simplify the constraints. The optimal value of  $\min_{y \in \mathcal{E}^\alpha} \langle \theta_{adv}, y \rangle$  is the objective value of the best (w.r.t.  $\theta_{adv}$ )  $\alpha$ -expansion of  $x^t$ . The best  $\alpha$ -expansion of a fixed solution  $x^t$  can be found by solving a minimum cut problem in



Table 1: Results from the output of (5) on three stereo vision instances. More details in Appendix F.

Instance	Costs changed	Weights changed	(normalized) Recovery error bound	$\ \hat{x}_V - \hat{x}_V^{MAP}\ _1/2n$
tsukuba	4.9%	2.3%	0.0518	0.0027
venus	22.5%	1.3%	0.0214	0.0016
cones	1.2%	2.1%	0.0437	0.0022

an auxiliary graph  $G_{aux}^{x^t}(\alpha)$  whose weights depend on linearly on the objective  $\theta_{adv}$ , and therefore depend linearly on our optimization variables  $(c, w)$  (Boykov et al., 2001, Section 4). Therefore, the optimization problem  $\min_{y \in \mathcal{E}^\alpha} \langle \theta_{adv}, y \rangle$  can be expressed as a minimum cut problem. Because this minimum cut problem can be written as a linear program, we can rewrite each constraint as

$$\langle \theta_{adv}, x^t \rangle \leq \min_{z: A(\alpha)z = b(\alpha), z \geq 0} \langle \theta_{adv}, z \rangle, \quad (4)$$

where  $\{A(\alpha)z = b(\alpha), z \geq 0\}$  is the feasible region of the standard metric LP corresponding to the minimum cut problem in  $G_{aux}^{x^t}(\alpha)$ . The number of vertices in  $G_{aux}^{x^t}(\alpha)$  and the number of constraints in  $A(\alpha)z = b(\alpha)$  is  $\text{poly}(m, n, k)$  for all  $\alpha$ . We now derive an equivalent linear formulation of (4) using a careful application of strong duality. The dual to the LP on the RHS is:

$$\underset{\nu}{\text{maximize}} \quad \langle b(\alpha), \nu \rangle, \text{ s.t. } A(\alpha)^T \nu \leq \theta_{adv}.$$

Because strong duality holds for this linear program, we have that (4) holds if and only if there exists  $\nu$  with  $A(\alpha)^T \nu \leq \theta_{adv}$  such that  $\langle \theta_{adv}, x^t \rangle \leq \langle b(\alpha), \nu \rangle$ .

This is a linear constraint in  $(c, w, \nu)$ . By using this dual witness trick for each label  $\alpha \in [k]$ , we obtain:

$$\begin{aligned} & \underset{c \geq 0, w \geq 0, \{\nu_\alpha\}}{\text{minimize}} && f(c, w) && (5) \\ & \text{subject to} && \langle \theta_{adv}, x^t \rangle \leq \langle b(\alpha), \nu_\alpha \rangle && \forall \alpha \\ & && A(\alpha)^T \nu_\alpha \leq \theta_{adv} && \forall \alpha. \end{aligned}$$

The constraints of (5) are linear in the optimization variables  $(c, w)$  and  $\nu_\alpha$ . The dimension of  $\theta_{adv}$  is  $nk + mk^2$ , so there are  $k(nk + mk^2 + 1)$  constraints and  $nk + m + \sum_\alpha |b(\alpha)| = \text{poly}(m, n, k)$  variables. Because minimization of the L1 distance  $f_1(c, w)$  can be encoded using a linear function and linear constraints, (5) is a linear program in this case. It is clear from the derivation of (5) that it is equivalent to (3). This proves Theorem 7.1. This formulation (5) can easily be input into ‘‘off-the-shelf’’ convex programming packages such as Gurobi (Gurobi Optimization, 2020).

## 8 Numerical results

Table 1 shows the results of running (5) on real-world instances of MAP inference to find nearby  $(2, 1, \psi)$ -

expansion stable instances. We study *stereo vision* models using images from the Middlebury stereo dataset (Scharstein & Szeliski, 2002) and Potts models from Tappen & Freeman (2003). Please see Appendix F for more details about the models and experiments.

We find, surprisingly, that only relatively sparse changes are required to make the observed instances  $(2, 1, \psi)$ -expansion stable with  $\psi = 1$ . On these instances, we evaluate the recovery guarantees by our bound from Theorem 5.2 and compare it to the actual value of the recovery error  $\|\hat{x} - \hat{x}^{MAP}\|_1/2n$ . In all of our experiments, we choose the *target* solution  $x^t$  for (5) to be equal to the MAP solution  $\hat{x}^{MAP}$  of the observed instance. Therefore, we find a  $(2, 1, \psi)$ -expansion stable instance that has the same MAP solution as our observed instance. The recovery error bound given by Theorem 5.2 is then also a bound for the recovery error between  $\hat{x}$  and  $\hat{x}^{MAP}$ , because  $\hat{x}^{MAP} = x^t$ . On these instances, the bounds from our curvature result (Theorem 5.2) are reasonably close to the actual recovery value. However, this bound uses the property that  $\hat{x}$  has good objective in the stable instance and so it is still a ‘‘data-dependent’’ bound in the sense that it uses an empirically observed property of the LP solution  $\hat{x}$ . In Appendix F, we show how to refine Corollary 5.3 to give non-vacuous recovery bounds that do not depend on  $\hat{x}$ .

## 9 Conclusion

We studied the phenomenon of near persistence of the local LP relaxation on instances of MAP inference in ferromagnetic Potts model. We gave theoretical results, algorithms (for finding nearby stable instances) and empirical results to demonstrate that even after a  $(2, 1, \psi)$ -perturbation stable instance is corrupted with noise, the solution to the LP relaxation is nearly persistent i.e., it approximately recovers the MAP solution. Our theoretical results imply that the local LP approximately recovers the MAP solution when the observed instance is close to a stable instance. Our empirical results suggest that real-world instances are very close to stable instances. These results together suggest a new explanation for the near-persistency of the solution of the local LP relaxation for MAP inference in practice.

## Acknowledgments

We would like to thank all the reviewers for their feedback. This work was supported by NSF AitF awards CCF-1637585 and CCF-1723344. Aravind Reddy was also supported in part by NSF CCF-1955351 and HDR TRIPODS CCF-1934931. Aravindan Vijayaraghavan was also supported by NSF CCF-1652491.

## References

- Abbe, E. Community detection and stochastic block models. *Foundations and Trends® in Communications and Information Theory*, 14(1-2):1–162, 2018. ISSN 1567-2190. doi: 10.1561/01000000067.
- Angelidakis, H., Makarychev, K., and Makarychev, Y. Algorithms for stable and perturbation-resilient problems. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 438–451, 2017.
- Archer, A., Fakcharoenphol, J., Harrelson, C., Krauthgamer, R., Talwar, K., and Tardos, É. Approximate classification via earthmover metrics. In *Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1079–1087. Society for Industrial and Applied Mathematics, 2004.
- Bilu, Y. and Linial, N. Are stable instances easy? In *Innovations in Computer Science*, pp. 332–341, 2010.
- Birchfield, S. and Tomasi, C. A pixel dissimilarity measure that is insensitive to image sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(4):401–406, 1998.
- Boykov, Y., Veksler, O., and Zabih, R. Fast approximate energy minimization via graph cuts. *IEEE Transactions on pattern analysis and machine intelligence*, 23(11):1222–1239, 2001.
- Chekuri, C., Khanna, S., Naor, J. S., and Zosin, L. Approximation algorithms for the metric labeling problem via a new linear programming formulation. In *Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA ’01, pp. 109–118, USA, 2001. ISBN 0898714907.
- Dahlhaus, E., Johnson, D. S., Papadimitriou, C. H., Seymour, P. D., and Yannakakis, M. The complexity of multiway cuts. In *Proceedings of the twenty-fourth annual ACM symposium on Theory of computing*, pp. 241–251, 1992.
- Globerson, A. and Jaakkola, T. S. Fixing max-product: Convergent message passing algorithms for map lp-relaxations. In *Advances in neural information processing systems*, pp. 553–560, 2008.
- Globerson, A., Roughgarden, T., Sontag, D., and Yildirim, C. How hard is inference for structured prediction? volume 37 of *Proceedings of Machine Learning Research*, pp. 2181–2190, Lille, France, 07–09 Jul 2015. PMLR. URL <http://proceedings.mlr.press/v37/globerson15.html>.
- Guédon, O. and Vershynin, R. Community detection in sparse networks via grothendieck’s inequality. *Probability Theory and Related Fields*, 165(3-4):1025–1049, 2016.
- Gurobi Optimization, L. Gurobi optimizer reference manual, 2020. URL <http://www.gurobi.com>.
- Kleinberg, J. and Tardos, E. Approximation algorithms for classification problems with pairwise relationships: Metric labeling and markov random fields. *Journal of the ACM (JACM)*, 49(5):616–639, 2002.
- Kolmogorov, V. and Wainwright, M. J. On the optimality of tree-reweighted max-product message-passing. In *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*, pp. 316–323, 2005.
- Kovtun, I. Partial optimal labeling search for a np-hard subclass of (max,+) problems. In *Joint Pattern Recognition Symposium*, pp. 402–409. Springer, 2003.
- Lang, H., Sontag, D., and Vijayaraghavan, A. Optimality of approximate inference algorithms on stable instances. In *International Conference on Artificial Intelligence and Statistics*, pp. 1157–1166, 2018.
- Lang, H., Sontag, D., and Vijayaraghavan, A. Block stability for map inference. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 216–225, 2019.
- Makarychev, K., Makarychev, Y., and Vijayaraghavan, A. Sorting noisy data with partial information. In *Proceedings of the 4th conference on Innovations in Theoretical Computer Science*, pp. 515–528. ACM, 2013.
- Makarychev, K., Makarychev, Y., and Vijayaraghavan, A. Bilu–linial stable instances of max cut and minimum multiway cut. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pp. 890–906. SIAM, 2014.
- McSherry, F. Spectral partitioning of random graphs. In *Proceedings of the 42nd IEEE symposium on Foundations of Computer Science*, FOCS ’01, pp. 529–, Washington, DC, USA, 2001. IEEE Computer Society. ISBN 0-7695-1390-5. URL <http://dl.acm.org/citation.cfm?id=874063.875554>.
- Meshi, O., London, B., Weller, A., and Sontag, D. Train and test tightness of lp relaxations in structured prediction. *The Journal of Machine Learning Research*, 20(1):480–513, 2019.

- Roughgarden, T. (ed.). *Beyond the Worst-Case Analysis of Algorithms*. Cambridge University Press, 2021. doi: 10.1017/9781108637435.
- Rowland, M., Pacchiano, A., and Weller, A. Conditions beyond treewidth for tightness of higher-order lp relaxations. In *Artificial Intelligence and Statistics*, pp. 10–18, 2017.
- Scharstein, D. and Szeliski, R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International journal of computer vision*, 47(1):7–42, 2002.
- Shekhovtsov, A. Exact and partial energy minimization in computer vision. 2013.
- 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.
- Sontag, D. *Approximate Inference in Graphical Models using LP Relaxations*. PhD thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, 2010.
- Sontag, D., Globerson, A., and Jaakkola, T. Introduction to dual decomposition for inference. In *Optimization for Machine Learning*. MIT Press, 2011.
- 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.
- 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.
- Thapper, J. et al. The power of linear programming for valued csps. In *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pp. 669–678. IEEE, 2012.
- Vershynin, R. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.
- Wainwright, M. J. and Jordan, M. I. *Graphical models, exponential families, and variational inference*. Now Publishers Inc, 2008.
- Wainwright, M. J., Jaakkola, T. S., and Willsky, A. S. Map estimation via agreement on trees: message-passing and linear programming. *IEEE transactions on information theory*, 51(11):3697–3717, 2005.
- Weller, A., Rowland, M., and Sontag, D. Tightness of lp relaxations for almost balanced models. In *Artificial Intelligence and Statistics*, pp. 47–55, 2016.