SDP Gaps and UGC Hardness for Multiway Cut, 0-Extension and Metric Labelling

Rajsekar Manokaran[†] [*] Joseph (Seffi) Naor[‡] Prasad Raghavendra[♯†]
Roy Schwartz[‡]

[†] Department of Computer Science,
Princeton University.
Princeton, NJ.
rajsekar@cs.princeton.edu

[‡] Department of Computer Science,
Technion,
Haifa, Israel.
{naor,schwartz}@cs.technion.ac.il

[♯] Department of Computer Science
and Eng.,
University of Washington,
Seattle, WA.
prasad@cs.washington.edu

Abstract

The connection between integrality gaps and computational hardness of discrete optimization problems is an intriguing question. In recent years, this connection has prominently figured in several tight UGC-based hardness results. We show in this paper a direct way of turning integrality gaps into hardness results for several fundamental classification problems. Specifically, we convert linear programming integrality gaps for the Multiway Cut, 0-Extension and Metric Labeling problems into UGC-based hardness results. Qualitatively, our result suggests that if the unique games conjecture is true then a linear relaxation of the latter problems studied in several papers (so-called *earthmover* linear program) yields the best possible approximation. Taking this a step further, we also obtain integrality gaps for a semi-definite programming relaxation matching the integrality gaps of the earthmover linear program. Prior to this work, there was an intriguing possibility of obtaining better approximation factors for labeling problems via semi-definite programming.

1 Introduction

The connection between integrality gaps and computational hardness of discrete optimization problems is an intriguing question. For a given problem, an integrality gap provides a bound on the approximation factor that can be obtained from rounding a specific linear programming or semi-definite programming formulation of the problem. In contrast, a computational hardness result applies to all polynomial time algorithms.

Yet, it has often transpired for many problems that hardness results turn out to match the best integrality gaps known (e.g., MAX-3SAT, maximum cut, vertex cover, etc.). For example, in the case of the maximum cut problem, the search for the optimal hardness result was directly inspired and aided by the known integrality gaps for the standard semi-definite relaxation of the problem. Thus, researchers are left to wonder if indeed there is a direct relationship between integrality gaps and hardness results.

In this paper we show for a set of labeling problems a direct way of turning integrality gaps into hardness results, assuming the unique games conjecture (UGC). In particular, we show how to convert integrality gaps for a linear programming relaxation of the metric labeling problem (and special cases of it) into a (conditional) hardness result.

The metric labeling problem falls under the class of edge deletion problems along with many other classic optimization problems. In an edge deletion problem, given an undirected graph $G = (V, E)$ and a non-negative weight function w on E, the goal is to find a minimum weight set of edges E' such that $G' = (V, E - E')$ satisfies certain properties. A special case is when the set of deleted edges forms a *cut*. The simplest and probably most

[*]Supported by NSF grants MSPA-MCS 0528414, and ITR 0205594

[†]This work was done while the author was visiting Princeton University and Microsoft Research.

familiar problem in this class is the minimum (s,t) cut problem. Given two terminals s and t, the goal is to find a minimum weight cut that separates s and t. This problem can be solved precisely in polynomial time following the classic work of Ford and Fulkerson.

The Multiway Cut problem is a natural generalization of the minimum (s,t) cut problem when more than two terminals are involved. The input is a set of k terminals $T \subseteq V$ and the goal is to find a minimum weight set of edges that separates every pair of terminals. This problem is NP-hard and a $(2-\frac{2}{k})$-approximation algorithm that uses the classic (s,t)-cut algorithm as a subroutine is known [9]. Based on a novel geometric relaxation, Calinescu et al. [8] obtained a $\frac{3}{2}-\frac{1}{k}$ approximation for the problem. Continuing this line of work, [13] obtained tight integrality gaps for the case $k=3$, and improved approximation factors for general k (about 1.3438).

The 0-Extension problem [15, 16] is a generalization of the Multiway Cut problem in which a metric d is defined on the terminal set T. The goal is to assign to each vertex $v \in V$ a terminal $t(v)$ in T, while minimizing the total cost given by $\sum_{u,v\in E} w(u,v)d(t(u),t(v))$. In case the metric on the terminals T is the uniform metric, then the problem reduces to the Multiway Cut problem. Calinescu et al. [5] obtained an $O(\log|T|)$-approximation algorithm for 0-Extension. The approximation factor was improved to $O(\log|T|/\log\log|T|)$ in [10] using a better analysis. The ideas from the 0-Extension problem [5, 10] have found further applications in metric embeddings [23] and in analysis[24].

Motivated by applications in computer vision, Kleinberg et al. [22] introduced the Metric Labeling (ML) problem. The input to the Metric Labeling problem is a metric space (T,d) of *labels* and a non-negative cost function c on vertex-label pairs. The objective is to find an assignment of labels to the vertices minimizing $\sum_{v\in V} c(v,t(v)) + \sum_{(u,v)\in E} w(u,v)\ d(t(u),t(v))$. This is a generalization of the 0-Extension problem, since the assignment cost function c is arbitrary. Using an approximate representation of metrics as a combination of dominating tree metrics [4], [22] gave an approximation algorithm for Metric Labeling. Its approximation factor can be shown to be $O(\log|T|)$ using the later improvement of [11] in embedding metrics into dominating tree metrics. A special case of Metric Labeling that is of particular interest is the Uniform Metric Labeling (UML) problem. Here the distance metric d on the labels T is just the uniform metric, i.e., $d(\ell_1,\ell_2)=1$ for all labels $\ell_1 \neq \ell_2$. For Uniform Metric Labeling a factor 2 approximation algorithm is known [22]. Constant factor approximation algorithms [22, 12, 6, 1] are known for several other special cases of metrics.

Inspired by the geometric relaxation for the Multiway Cut problem, Chekuri et al. [6] proposed an earthmover metric linear relaxation for the Metric Labeling and 0-Extension problems. They also showed that the integrality gap of the earthmover relaxation is at least as good as the approximation factor of the Kleinberg-Tardos algorithm [22] for general metrics. Archer et al. [1] gave an earthmover relaxation based Metric Labeling algorithm whose performance depends on the decomposability of the metric d. However, even the earthmover linear relaxation proved unsuccessful in obtaining approximation factors better than $O(\log|T|)$ for Metric Labeling. In fact, for the problems of Metric Labeling and 0-Extension, integrality gaps of $\Omega(\log|T|)$ and $\Omega((\log|T|)^{\frac{1}{2}})$, respectively, were shown for the earthmover relaxation [14].

Nevertheless, the hardness results known for Metric Labeling and 0-Extension still do not match the best known approximation algorithms. Chuzhoy and Naor [7] showed that for any $\epsilon > 0$ there is no polynomial time algorithm that approximates the Metric Labeling problem within a factor of $O((\log|T|)^{\frac{1}{2}-\epsilon})$, unless $NP \subseteq DTIME(n^{poly(\log n)})$. Building on the work of [7], [14] showed that there is no polynomial time algorithm that approximates 0-Extension within a factor of $O((\log|T|)^{\frac{1}{4}-\epsilon})$, unless $NP \subseteq DTIME(n^{poly(\log n)})$. Even in the case of the Multiway Cut problem nothing better than APX-hardness [9] is known.

In the above discussion, an intriguing possibility that remains open is the use of semidefinite programming (SDP) to obtain better approximation factors for Metric Labeling. In fact, the earthmover relaxation has a natural semidefinite counterpart. Even for the case of Multiway Cut, obtaining a better approximation using semidefinite programming has not been ruled out.

1.1 Results

Roughly speaking, we show that semidefinite programming relaxations do not yield better approximation factors than the earthmover linear program. Towards this, we harness the connection between the Unique Games Conjecture (UGC) [17] and semidefinite programming. The interplay between UGC based hardness results and SDP integrality gaps has been very fruitful in recent years [21, 18, 19, 2, 3].

Our approach is as follows: Starting with an integrality gap instance Π for the earthmover linear program, we show a matching unique games based hardness. By using this hardness reduction, along with a well known SDP gap for Unique Games [21], we obtain corresponding SDP gaps. The SDP gap obtained is exactly equal to the gap of the earthmover linear program instance Π, with which we began the reduction. Thus, we obtain both SDP integrality gaps and unique games based hardness results, matching the earthmover linear program integrality gap.

We implement this approach for all of the above problems: Multiway Cut, 0-Extension and Metric Labeling. More precisely, we prove:

Theorem 1.1 *Assuming the Unique Games Conjecture, for the Multiway Cut, the 0-Extension, and the Metric Labeling problems, the following holds: Given an instance $\mathcal{H}$ with integrality gap α for the earthmover linear program, there is a NP-hardness reduction showing that the problem cannot be approximated to a factor better than α. Further, the instances produced by the reduction have the same set of labels as $\mathcal{H}$.*

The following theorem refers to the earthmover semidefinite relaxation appearing in Section 2.1.

Theorem 1.2 *For the Multiway Cut, the 0-Extension, and the Metric Labeling problems, the integrality gap of the earthmover semidefinite relaxation(EM-SDP) is equal to the integrality gap of the earthmover linear program.*

As the reduction always produces an instance with the same set of labels, the following stronger result holds:

Theorem 1.3 *Assuming the Unique Games Conjecture, it is NP-hard to approximate the Metric Labeling and 0-Extension problems with any finite metric (T, d) to a factor better than the integrality gap of the earthmover linear program on (T, d).*

Note that determining the exact value of the earthmover linear program integrality gap for these problems is not always easy. The following table shows the earthmover linear program gaps and the best known approximation factors.

Problem	**Integrality Gap**	**App. Factor**
3-way cut	12/11 [13]	12/11 [13]
0-Extension	$\Omega\big((\log \lvert T\rvert)^{\frac{1}{2}}\big)$ [14]	$O\big(\frac{\log \lvert T\rvert}{\log\log \lvert T\rvert}\big)$ [10]
ML	$\Omega(\log \lvert T\rvert)$ [14]	$O(\log \lvert T\rvert)$ [22]
UML	$2 - \frac{2}{\lvert T\rvert}$ [22]	2 [22]

The reductions in this paper produce instances whose size is at least doubly exponential in the size of the earthmover linear program integrality gap instance. Therefore, the hardness results as well as the SDP gaps hold only for fixed values of $|T|$. In other words, the results of this work apply for constant factor gaps and hardness results.

Interestingly, the reductions in this paper would apply even if the distance function between the labels does not satisfy triangle inequality. In particular, it is enough that $d(x, x) = 0$ and $d(x, y) \neq 0$ for $x \neq y$.

In a somewhat different direction, Raghavendra [26] obtains UG-hardness for every CSP (constraint satisfaction problem), starting from an appropriate SDP integrality gap. In particular, this general conversion from SDP gaps to UG-hardness applies to the problems Metric Labeling, 0-Extension and Multiway Cut. However, the reduction in [26] makes crucial use of the SDP vectors, and thus would not apply to linear programming integrality gaps. Although both [26] and our work proceed by converting integrality gaps to hardness results, the soundness proofs are very different. For all the problems in this work, the objective is to minimize the number of edges cut. Hence, along the lines of many other UG hardness results for cut problems [21, 20], the proof uses noise stability of functions.

1.2 Proof Overview

To illustrate the main ideas, we outline the reduction for the 3-way cut problem. Let $G = (V, E)$ be a 3-way cut instance with terminals $\{t_1, t_2, t_3\}$. The following is the earthmover linear programming relaxation for the problem:

$$
\begin{array}{ll}
\text{Minimize} & \frac{1}{2} \sum_{e=(u,v)\in E} \|X_u - X_v\|_1 \\
\text{subject to:} & X_u^{(1)} + X_u^{(2)} + X_u^{(3)} = 1 \quad \forall u \in V \\
& X_u^i \geqslant 0 \\
& X_{t_1} = (1,0,0), X_{t_2} = (0,1,0), X_{t_3} = (0,0,1)
\end{array}
$$

A crucial ingredient in all UG hardness reductions is a Dictatorship Test. A function $F : \{1,2,3\}^R \to \{1,2,3\}$ is said to be a *dictator* if the function is given by $F(x) = x_i$ for some fixed i. The input to a dictatorship test consists of a function $F : \{1,2,3\}^R \to \{1,2,3\}$. The objective is to query the function F at a few locations, and distinguish whether the function is a dictator or far from every dictator.

Given a dictatorship test, the UG hardness reduction usually follows by standard techniques. Roughly speaking, one introduces a vertex for every point in $\{1,2,3\}^R$ and translates the queries made by the dictatorship test in to constraints between these vertices. The resulting gadget is usually referred to as the "Long Code Gadget".

Making things concrete, we shall now describe the long code gadget used as part of our reduction. Actually, we convert an integrality gap instance for the Earthmover LP in to a long code gadget.

Let us suppose $G = (V, E)$ is an integrality gap instance for the above linear program. Let $\mathsf{LP}(G)$ and $\mathsf{OPT}(G)$ denote the optimal LP and integral values, respectively. The LP solution associates each vertex v in V with a point X_v on the 3-dimensional simplex. The coordinates of X_v can be thought of as probabilities of assigning the corresponding labels.

From G, we shall construct a 3-cut instance G' such that :

- There exist special 3-way cuts in G' whose cost equals the linear programming optimum $\mathsf{LP}(G)$. These cuts will be referred to as *dictator* cuts.
- A 3-way cut solution in G' which is far from every dictator cut pays at least the integral optimum $\mathsf{OPT}(G)$.

The vertices of G' are as follows : For each vertex v of G introduce a group Ω_v^R of 3^R vertices. The vertices in Ω_v^R are indexed by vectors $\{1,2,3\}^R$. It is useful to think of Ω_v^R as having a product probability distribution X_v^R on it.

For example, consider the terminal t_1 of the 3-way cut instance G. The corresponding LP assignment X_{t_1} is a corner of the simplex $e_1 = (1,0,0)$. Hence the probability distribution $X_{t_1}^R$ is non-zero on a single vertex $(1,1,1,\ldots,1)$. Similarly for each t_i, the probability distribution $X_{t_i}^R$ on $\Omega_{t_i}^R$ is nonzero only at $(i,i,\ldots,i)$. These special vertices are the terminals of G'. More precisely, the terminals of G' are the vertices $(i,i,\ldots,i) \in \Omega_{t_i}^R$.

A 3-way cut solution assigns to each vertex a label from the set $\{1,2,3\}$. Thus a 3-way cut solution to G' consists of a set of functions $F_v : \Omega_v^R \to \{1,2,3\}$. There are two special 3-way cut solutions that will be of interest:

- The set of functions $F_v(x) = x_i$ for some i. These functions form a feasible 3-cut solution, since they assign different labels to all the terminals. We shall refer to these solutions as *dictator* cuts.
- Each function F_v is a constant function. These solutions will be referred to as *integral cuts*

For an edge $e = (v,w)$ in the graph G, we will introduce edges between groups Ω_v^R and Ω_w^R. The edges introduced are such that the *dictator* cuts have a cost close to $\mathsf{LP}(G)$. We illustrate the basic idea with an example. Let $e = (v,w)$ be an edge in G, with $X_v = (\frac{1}{3}, \frac{1}{2}, \frac{1}{6})$ $X_w = (\frac{1}{6}, \frac{1}{2}, \frac{1}{3})$. The edges between groups Ω_v^R and Ω_w^R are given by a joint distribution over pairs $x \in \Omega_v^R$, $y \in \Omega_w^R$. Generate each coordinate of x according to the probability distribution X_v. To generate y, we shall mimic the flow of probability mass required to convert distribution X_v into X_w. Specifically, the i^{th} coordinate y_i is generated from x_i using the following distribution:

> If $x_i = 1$, then $y_i = 1$ with probability $\frac{1}{2}$ and $y_i = 3$ with the remaining probability. If $x_i = 2$ or 3, then $y_i = x_i$.

It is easy to check that if x_i is generated according to distribution $X_v = (\frac{1}{3}, \frac{1}{2}, \frac{1}{6})$, then the distribution of y_i is same as $X_w = (\frac{1}{6}, \frac{1}{2}, \frac{1}{3})$.

Consider a *dictator* cut given by functions $F_v(x) = x_1$ and $F_w(y) = y_1$. The cost of the cut is equal to the probability that $x_1 \neq y_1$ when x, y are generated as above. But this is exactly equal to the total probability mass that *flows* so as to change distribution X_v to X_w. In this case, the probability of $x_1 \neq y_1$, is $\frac{1}{6} = \frac{1}{2}\|X_v - X_w\|_1$. Consequently, the *dictator* cuts pay exactly the LP value $\mathsf{LP}(G)$.

In an *integral* cut, the group of $[3]^R$ vertices corresponding to a vertex v all have the same label. Intuitively, an *integral* 3-way cut is assigning a label to the vertex v in the original graph G. In fact, an *integral* 3-way cut of G' corresponds to a 3-way cut of G. Thus, if all the functions F_v were constant functions, then the cost of the cut is at least the minimum cost $\mathsf{OPT}(G)$ of a 3-way cut of G.

We need to ensure that the functions which are far from a *dictator* function have a cost at least the integral optimum $\mathsf{OPT}(G)$. Towards this end, we shall introduce *noise sensitivity* edges. Inside each group Ω_v^R we will introduce edges between pairs (x, y) where x is from the distribution X_v^R, and y is generated by perturbing the coordinates of x. By an appropriate choice of parameters, the total cost of *noise sensitivity* edges overwhelms the remaining edges. Using results on noise stability, if a function $F_v : \Omega_v^R \to \{1, 2, 3\}$ cuts a small fraction of the *noise sensitivity* edges, then either:

- The function F_v is close to a dictator function (more precisely, it has an influential variable).
- Function F_v is close to a constant function.

Hence, either we obtain a function F_v with an influential variable, or the cost of the cut is $\mathsf{OPT}(G)$. Using standard techniques, such a gadget can be used to obtain a unique games based hardness result.

2 Preliminaries

For a positive integer k, Δ_k denotes the the k dimensional simplex. The notation $[k]$ refers to the set $\{1, \ldots, k\}$. From [6], without loss of generality, it can be assumed that the assignment costs in Metric Labeling are either zero or infinity. Thus, Metric Labeling with assignment costs in $\{0, \infty\}$ is called *restricted* Metric Labeling. Towards setting up notation, we define the problems Multiway Cut, 0-Extension and Metric Labeling below.

Definition 2.1 *An instance of the (restricted)* ML *problem is a weighted graph, $\mathcal{H} = (V(\mathcal{H}), E(\mathcal{H}), w_e)$, along with a set of labels $\mathcal{L}$, a family of subsets $\{\mathcal{L}(v)\}_{v \in V(\mathcal{H})}$, and a metric d on $\mathcal{L}$. A valid labeling is a mapping $\Lambda : V(\mathcal{H}) \to \mathcal{L}$ such that for each vertex, $v \in V(\mathcal{H})$, $\Lambda(v)$ belongs to $\mathcal{L}(v)$. The cost of a labeling Λ, $\mathrm{Val}_\Lambda(\mathcal{H})$, is*

$$\sum_{(u,v)=e \in E(\mathcal{H})} w_e \, d(\Lambda(u), \Lambda(v)).$$

The value of the instance, $\mathsf{OPT}(\mathcal{H})$, is the minimum cost labeling for the instance.

Definition 2.2 *An instance of* Multiway Cut *problem consists of a weighted graph, $\mathcal{H} = (V(\mathcal{H}), E(\mathcal{H}), w_e)$, along with a set of terminals $\mathcal{L} \subset V(\mathcal{H})$. The objective is to remove a set of edges of minimum weight so as to separate every pair of terminals.*

The Multiway Cut problem can be formulated as a labeling problem (with a uniform metric) as follows: A valid multiway cut corresponds to a labeling $\Lambda : V(\mathcal{H}) \to \mathcal{L}$ such that for each terminal $t \in \mathcal{L}$, $\Lambda(t) = t$. The cost of such a labeling Λ, $\mathrm{Val}_\Lambda(\mathcal{H})$ is given by $\sum_{(u,v) \in E(\mathcal{H}), \Lambda(u) \neq \Lambda(v)} w_e$. The value of the instance $\mathrm{Val}(\mathcal{H})$ is the minimum cost labeling for the instance.

Definition 2.3 *An instance of* 0-Extension *problem consists of a weighted graph, $\mathcal{H} = (V(\mathcal{H}), E(\mathcal{H}), w_e)$, along with a set of terminals $\mathcal{L} \subset V(\mathcal{H})$ with a metric d on them. The objective is to assign each vertex v to a terminal $\Lambda(v) \in \mathcal{L}$ such that the following cost is minimized:*

$$\sum_{(u,v)=e \in E(\mathcal{H})} w_e \, d(\Lambda(u), \Lambda(v)).$$

The value of the instance, $\mathrm{Val}(\mathcal{H})$ *is the minimum cost labeling for the instance.*

2.1 Earthmover Linear Program for Metric Labeling

The following linear programming relaxation for ML was introduced in [6]. Let $\mathcal{H} = (V(\mathcal{H}), E(\mathcal{H}), w_e)$ be an instance of metric labeling. Intuitively, the LP asks for an embedding of the vertices $V(\mathcal{H})$ on the k dimensional simplex Δ_k. For every vertex v, the corresponding point $X_v \in \Delta_k$ represents the probability distribution of each label being assigned to v. For example, each corner of the simplex represents a particular label. The labeling constraint $\Lambda(v) \in \mathcal{L}(v)$ is enforced by a linear constraint on the probability distribution X_v. These labeling constraints force the point X_v to lie in the face containing the allowed labels $\mathcal{L}(v)$, denoted by $\Delta_{\mathcal{L}(v)}$. The objective is to minimize the weighted sum of the *earthmover* distance between adjacent vertices.

Definition 2.4 (Earthmover Distance) *Given two points $X, Y \in \Delta_k$, and a metric $d(i,j)$ on $[k]$, the earthmover distance, $d_{\bowtie}(X,Y)$ is given by the optimal value of the following LP:*

$$\begin{aligned} &\textit{Minimize} && \sum_{i,j} d(i,j) f_{ij} \\ &\textit{s.t.} && \sum_i f_{ij} = Y_j \qquad \sum_j f_{ij} = X_i \qquad \forall i,j \in [k] \\ &&& f_{ij} \geqslant 0 \end{aligned}$$

In other words, the earthmover distance is the minimum cost of moving the probability mass from distribution X to Y, given the distance metric d on the labels. It is easy to see that this defines a metric on the simplex Δ_k. Thus, the earthmover distance generalizes a metric on k points to a metric on Δ_k such that the distance between corner points is the same as the original metric. In this notation, the linear program of [6] is simply:

$$\boxed{\begin{aligned} &\text{Minimize} && \sum_{(u,v) \in E(\mathcal{H})} w_e d_{\bowtie}(X_u, X_v) \\ &&& \qquad\qquad\qquad \text{(EM-LP)} \\ &\text{s.t.} && X_u \in \Delta_{\mathcal{L}(u)} \quad \forall u \in V(\mathcal{H}) \end{aligned}}$$

Our reduction starts with an instance of metric labeling along with an optimal solution to the linear program (EM-LP).

Definition 2.5 (Structured Integrality Gap) *A structured integrality gap instance to the metric labeling problem is a 4-tuple $(\mathcal{H}, \mathcal{L}, d, \{X_v\})$ where $X_v \in \Delta_k$ is an optimal solution to the LP* (EM-LP)*. In addition, the metric labeling instance satisfies the following properties:*

$$\sum_{e \in E(\mathcal{H})} w_e = 1 \textit{ and} \qquad\qquad d(i,j) \leqslant 1, \textit{ for all } 1 \leqslant i,j \leqslant k.$$

The following parameters will be important for the reduction: $m = |V(\mathcal{H})|$, $\alpha = \min_{a \in V(\mathcal{H}), X_a^i \neq 0} X_a^i$, $\beta = \min_{i \neq j} d(i,j)$, $\mathsf{LP}(\mathcal{H})$ is the value of the LP optimum and $\mathsf{OPT}(\mathcal{H})$ is the value of the integer optimum.

The following semidefinite program is a simple generalization of the earthmover metric linear program. Our goal is to show integrality gaps for this semidefinite program.

$$
\text{Minimize} \quad \sum_{(u,v)\in E(\mathcal{H})} w_e \sum_{i,j\in\mathcal{L}} d(i,j)(v_i \cdot v_j) \tag{EM-SDP}
$$

subject to:

$$
\begin{aligned}
&\sum_{i\in\mathcal{L}} ||v_i||_2^2 = 1 && \forall v \in V(\mathcal{H}) && (1)\\
&||v_i||_2^2 = 0 && \forall v \in V(\mathcal{H}), i \notin \mathcal{L}(v) && (2)\\
&v_i \cdot v_j = 0 && \forall v \in V(\mathcal{H}), i,j \in \mathcal{L} && (3)\\
&u_i \cdot v_j \geqslant 0 && \forall u,v \in V(\mathcal{H}), i,j \in \mathcal{L} && (4)\\
&u_i \cdot \sum_{j\in\mathcal{L}} v_j = ||u_i||_2^2 && \forall u,v \in V(\mathcal{H}), i \in \mathcal{L} && (5)
\end{aligned}
$$

2.2 Unique Games Conjecture, Long Codes, and Analytic Notions

To simplify the presentation, we will use the following version of the Unique Games Conjecture [20], which was shown to be equivalent to Khot's original conjecture [17].

Conjecture 2.6 (Unique Games Conjecture) *For any constant $\eta > 0$, there exists large enough constant R such that given a bipartite unique games instance $\Upsilon = (\mathcal{X} \cup \mathcal{Y}, E, \Pi = \{\pi_e : [R] \to [R] \mid e \in E\}, [R])$ with number of labels R, it is NP-hard to distinguish between the following two cases:*

- *There exists an assignment A of labels such that for $1-\eta$ fraction of the vertices $v \in X$, all the edges (v, w) are satisfied.*
- *No assignment satisfies more than a η-fraction of the constraints Π.*

Our reduction, like many previous reductions, will use long code gadgets. We recall a few standard definitions to facilitate our analysis. Let Ω denote a finite probability space with k different atoms. Let $\{\chi_0 = 1, \chi_1, \chi_2, \ldots, \chi_{k-1}\}$ be an orthonormal basis for the space $L_2(\Omega)$. For $\sigma \in [k]^R$, define $\chi_\sigma(x) = \prod_{i\in[R]} \chi_{\sigma_i}(x_i)$. Every function $F : \Omega^R \to \mathbb{R}$ can be expressed as a multilinear polynomial as follows:

$$
F(x) = \sum_\sigma \hat{F}(\sigma)\chi_\sigma(x).
$$

Definition 2.7 *For $0 \leqslant \rho \leqslant 1$, define the operator T_ρ on $L_2(\Omega^R)$ as $T_\rho F(x) = \mathbf{E}[F(y) \mid x]$ where $y_i = x_i$ with probability ρ and a random element from Ω with probability $1-\rho$. Formally,*

$$
T_\rho F(x) = \sum_{\sigma\in[k]^R} \rho^{|\sigma|}\hat{F}(\sigma)\chi_\sigma(x),
$$

where $|\sigma|$ is the number of non-zero coordinates in σ.

For a function $F : \Omega^R \to \mathbb{R}$ define the influence and low degree influence of the i^{th} coordinate as follows:

$$
\mathrm{Inf}_i(F) = \mathbf{E}_x[\mathbf{Var}_{x_i}[F]] = \sum_{\sigma_i\neq 0} \hat{F}^2(\sigma)
$$

$$
\mathrm{Inf}_i^{<t}(F) = \sum_{\sigma_i\neq 0, |\sigma|\leqslant t} \hat{F}^2(\sigma).
$$

Fact 2.8 *For a function F with* $\mathbf{Var}[F] \leqslant 1$,

$$\sum_i \mathrm{Inf}_i^{<t}(F) \leqslant t.$$

The Gaussian noise stability Γ_ρ is defined as follows:

Definition 2.9 *Given $\mu \in [0,1]$, let $t = \Phi^{-1}(\mu)$ where Φ denotes the distribution function of the standard Gaussian. Then,*

$$\Gamma_\rho(\mu) = \Pr[X \leqslant t, Y \leqslant t],$$

where (X, Y) is a two-dimensional Gaussian vector with covariance matrix $\begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$.

The following theorem on noise stability of functions over a product probability space is essentially a restatement of Theorem 4.4 in Mossel et al. [25]

Theorem 2.10 *Let Ω be a finite probability space with the least non-zero probability of an atom at least α. For every $\epsilon, \zeta > 0$ there exists t, τ such that the following holds: For every function $F : \Omega^R \to [0,1]$ with $\mu = \mathbf{E}[F]$ and $\mathrm{Inf}_j^{\leqslant t}(F) < \tau$ for all $j \in [R]$,*

$$\mathbf{E}_x\left[F T_{1-\epsilon} F\right] \leqslant \Gamma_{1-\epsilon}(\mu) + \zeta.$$

2.3 Edge Test Distribution

Every point $X \in \Delta_k$ corresponds to a probability distribution over $[k]$. For $X, Y \in \Delta_k$, let Ω_X, Ω_Y denote the corresponding probability spaces. The earthmover distance $d_{\bowtie}(X, Y)$ suggests a natural way to generate correlated random variables $(x, y) \in \Omega_X^R \times \Omega_Y^R$. Let f_{ij} be the optimal value of the earthmover LP as in Definition 2.4. Then, we have the following joint probability distribution:

$$\Pr[y_i = b \wedge x_i = a] = f_{ij}.$$

The constraints of the LP ensure that the f_{ij}'s form a valid joint probability distribution. We will denote such a correlated distribution by $x \sim_{XY} y$. For an edge $e = (u, v) \in E(\mathcal{H})$ and an LP solution $\{X_u\}$, we will denote by $x \sim_e y$ the corresponding correlated distribution.

3 The Reduction

In this section, we shall describe the reduction from Unique Games to Metric Labeling. The same reduction applies with minor changes for the Multiway Cut and 0-Extension problems.

Let $\Psi = (\mathcal{H}, \mathcal{L}, d, \{X_v\})$ be a structured integrality gap instance for Metric Labeling. Without loss of generality, we may assume that the set of labels $\mathcal{L} = [k]$. Let $\Upsilon = (\mathcal{X} \cup \mathcal{Y}, E, \Pi, [R])$ be a UG instance. Let $\epsilon > 0$ and let $\delta = \epsilon^{7/8}$. We construct a metric labeling instance $(\mathcal{G}, \mathcal{L}, d)$ as in Figure 1.

Proof of *(of Theorem 1.1):* For Metric Labeling the proof directly follows from Theorems 3.1 and 3.7. As stated, the instance produced by the reduction also has the same set of labels $\mathcal{L}$.

For 0-Extension and Multiway Cut, the instances produced by the reduction have too many terminals. Specifically, for every vertex $w \in \mathcal{Y}$, terminal $t \in \mathcal{L}$ and $x \in [k]^R$ there are k^R vertices of the form $(w, t, x) \in V(\mathcal{G})$. For every vertex (w, t, x), the set of allowed labels in $\mathcal{G}$ is just $\{t\}$. Using standard techniques the graph $\mathcal{G}$ can be modified into $\mathcal{G}'$ with the correct set of terminals.

Introduce a new vertex in $V(\mathcal{G}')$ for each label in $t \in \mathcal{L}$. These new vertices are the terminals for $\mathcal{G}'$. For every vertex (w, t, x) with $t \in \mathcal{L}$, introduce an edge of infinite(sufficiently high) cost between t and (w, t, x). A solution to the instance $\mathcal{G}'$ will not cut any of the edges of infinite cost. This simulates the constraint that (w, t, x) is assigned label t. $\square$

The vertices of $\mathcal{G}$ are $V(\mathcal{G}) = \mathcal{Y} \times V(\mathcal{H}) \times [k]^R$.
The set of allowed labels for a vertex $(y, a, x) \in V(\mathcal{G})$ is $\mathcal{L}(a)$.
The weight of an edge is the probability it is output by the following test:

- Pick $u \in \mathcal{X}$ at random, and two of its neighbors w_1, w_2 independently at random. Let π_1, π_2 denote the permutations on edges (u, w_1) and (u, w_2).
- With probability δ perform the edge test, otherwise (with probability $1 - \delta$) perform the vertex test.

Edge test: (with probability δ)

- Pick an edge $e = (a, b) \in E(\mathcal{H})$ with the probability distribution w_e.
- Sample $x \sim_e y$, $x \in \Omega_a^R$, $y \in \Omega_b^R$.
- Output the edge $(w_1, a, \pi_1(x)) \leftrightarrow (w_2, b, \pi_2(y))$.

Vertex test: (with probability $1 - \delta$)

- Pick a uniformly random vertex $a \in V(\mathcal{H})$.
- Sample $x \in \Omega_a^R$. Sample $y \in [k]^R$ such that $y_i = x_i$ with probability $1 - \epsilon$ and $y_i =$ a new sample from Ω_a otherwise.
- Output the edge $(w_1, a, \pi_1(x)) \leftrightarrow (w_2, a, \pi_2(y))$.

Figure 1: The reduction.

3.1 Completeness

Theorem 3.1 *For every $\epsilon, \eta > 0$, given a UG instance Υ that is $1 - \eta$ strongly satisfiable and a structured integrality gap instance Ψ, the value of the metric labeling instance $(\mathcal{G}, \mathcal{L}, d)$ obtained from the reduction is at most $(1 - \eta)(\epsilon^{7/8}\,\mathsf{LP}(\mathcal{H}) + \epsilon) + \eta$.*

Proof of L:et λ denote a labeling to the UG instance Υ. Consider the labeling Λ to $\mathcal{G}$ that sets $\Lambda(u, a, x) = x_{\lambda(u)}$. It is easy to check that Λ is a valid labeling for the instance $\mathcal{G}$. Fix $\delta = \epsilon^{7/8}$. Then, the cost of the labeling Λ is:

$$
\begin{aligned}
& \mathbf{E}_{v,w_1,w_2}\Bigg[\delta \cdot \mathbf{E}_{(a,b)\in E(\mathcal{H})} \\
& \qquad \mathbf{E}_{x,y}\Big[d(\Lambda(w_1, a, \pi_1(x)), \Lambda(w_2, b, \pi_2(y)))\Big] \\
& \qquad + (1-\delta)\cdot \mathbf{E}_{a\in V(\mathcal{H})} \\
& \qquad \mathbf{E}_{x\sim_{1-\epsilon} y}\Big[d(\Lambda(w_1, a, \pi_1(x)), \Lambda(w_2, a, \pi_2(y)))\Big]\Bigg] \\
= \; & \mathbf{E}_{v,w_1,w_2}\Bigg[\delta \cdot \mathbf{E}_{(a,b)\in E(\mathcal{H})}\mathbf{E}_{x,y}\Big[d(x_{\pi_1(\lambda(w_1))}, y_{\pi_2(\lambda(w_2))})\Big] \\
& \qquad + \; (1-\delta)\cdot \mathbf{E}_{a\in V(\mathcal{H})} \\
& \qquad \mathbf{E}_{x\sim_{1-\epsilon} y}\Big[d(x_{\pi_1(\lambda(w_1))}, y_{\pi_2(\lambda(w_2))})\Big]\Bigg].
\end{aligned}
$$

With probability $1 - \eta$ over the choice of vertex u, the unique games assignment λ satisfies all the edges incident at u. Let us refer to these vertices u as *good* vertices. For a *good* u, for all choices of w_1, w_2, $\pi_1(\lambda(w_1)) =$

$\pi_2(\lambda(w_2)) = \lambda(u)$. Thus the expected cost for a *good* vertex is given by

$$\begin{aligned}
&\delta \cdot \mathbf{E}_{(a,b)\in E(\mathcal{H})}\mathbf{E}_{x,y}\Big[d(x_{\lambda(u)}, y_{\lambda(u)})\Big] \\
&\qquad + (1-\delta)\cdot \mathbf{E}_{a\in V(\mathcal{H})}\mathbf{E}_{x\sim_{1-\epsilon} y}\Big[d(x_{\lambda(u)}, y_{\lambda(u)})\Big] \\
&\quad \leqslant\ \delta \cdot \mathsf{LP}(\mathcal{H}) + (1-\delta)\cdot \epsilon.
\end{aligned}$$

For an arbitrary vertex u, the expected cost is always bounded by 1 since all the distances are bounded by 1. Thus, the total cost of the labeling Λ is at most $(1-\eta)\cdot\Big(\delta\,\mathsf{LP}(\mathcal{H}) + \epsilon\Big) + \eta$. □

Corollary 3.2 *For every $\gamma > 0$, there exists $\epsilon, \eta > 0$ such that, the value of the metric labeling instance $(\mathcal{G}, \mathcal{L}, d)$ obtained as in Theorem 3.1 is at most $\epsilon^{7/8}\,\mathsf{LP}(\mathcal{H})(1+\gamma)$.*

Proof of S:etting $\epsilon < (\gamma\,\mathsf{LP}(\mathcal{H})/2)^8$ and $\eta < \epsilon^{7/8}\,\mathsf{LP}(\mathcal{H})\gamma/4$ in Theorem 3.1 gives the required result. □

3.2 Soundness

Let Λ be a labeling of $(\mathcal{G}, \mathcal{L}, d)$ obtained from the reduction. Let ϵ be as defined in the reduction and let $\delta = \epsilon^{7/8}$. For $w \in \mathcal{Y}$, $a \in V(\mathcal{H})$, define k functions $F^i_{w,a} : [k]^R \to [0,1]$:

$$F^i_{w,a}(x) = \begin{cases} 1 & \text{if } \Lambda(w,a,x) = i \\ 0 & \text{otherwise} \end{cases}$$

For $u \in \mathcal{X}$, $a \in V(\mathcal{H})$, define k functions $G^i_{u,a} : [k]^R \to [0,1]$:

$$G^i_{u,a}(x) = \mathbf{E}_{w\in N(u)}\left[F^i_{w,a}(\pi_{uw}(x))\right].$$

Observe that for any x,

$$\begin{aligned}
\sum_{i=1}^{k} G^i_{u,a}(x) &= \sum_{i=1}^{k} \mathbf{E}_{w\in N(u)}\left[F^i_{w,a}(\pi_{uw}(x))\right] \\
&= \mathbf{E}_{w\in N(u)}\left[\sum_{i=1}^{k} F^i_{w,a}(\pi_{uw}(x))\right] = 1.
\end{aligned}$$

Define $\mu^i_{u,a} = \mathbf{E}_x[G^i_{u,a}(x)]$ where x is distributed according to the probability distribution of Ω_a^R. Further, define $\mu_{u,a} = (\mu^1_{u,a}, \ldots, \mu^k_{u,a})$. Hence, for all u, a we have

$$\sum_{i=1}^{k} \mu^i_{u,a} = \sum_{i=1}^{k} \mathbf{E}_x[G^i_{u,a}(x)] = \mathbf{E}_x\left[\sum_{i=1}^{k} G^i_{u,a}(x)\right] = 1.$$

Thus, $\mu_{u,a} \in \Delta_k$, i.e., it defines an embedding in the simplex. We will drop u and a when they are clear from the context.

For a vertex $u \in \mathcal{X}$, let $\mathrm{Val}^{\mathsf{Edge}}_\Lambda(u)$ and $\mathrm{Val}^{\mathsf{Vertex}}_\Lambda(u)$ denote the expected cost incurred by the edge and vertex tests respectively when the verifier chooses vertex u. We can write the cost of the labeling as follows.

$$\mathrm{Val}_\Lambda(\mathcal{G}) = \mathbf{E}_u\left[\delta\ \mathrm{Val}^{\mathsf{Edge}}_\Lambda(u) + (1-\delta)\,\mathrm{Val}^{\mathsf{Vertex}}_\Lambda(u)\right].$$

We will show that for most choices $u \in \mathcal{X}$, $a \in V(\mathcal{H})$, either the functions $G^i_{u,a}$ have an influential variable or they are close to constant functions. More precisely, we show that if the functions are neither constant nor have influential variables, then the cost of the vertex test on u, a is overwhelmingly large.

Lemma 3.3 *Fix $u \in \mathcal{X}$, $a \in V(\mathcal{H})$ and let G^i denote the family of functions associated with (u,a). For every ϵ we have*

$$\begin{aligned}\mathbf{E}_{x\sim_{(1-\epsilon)}y}\Big[\sum_{i,j\in\mathcal{L}} d(i,j)G^i(x)G^j(y)\Big] \\ \geqslant\ \beta\sum_i\Big(\mathbf{E}_x[G^i]-\mathbf{E}_x[G^iT_{1-\epsilon}(G^i)]\Big),\end{aligned}$$

where $\beta = \min_{i\neq j} d(i,j)$. Further, for all $\epsilon,\zeta > 0$, there exists t,τ such that if $\mathrm{Inf}_j^{<t}(G^i) < \tau$ for all $i \in [k], j \in [R]$, then

$$\begin{aligned}\mathbf{E}_{x\sim_{1-\epsilon}y}\Big[\sum_{i,j\in\mathcal{L}} d(i,j)G^i(x)G^j(y)\Big] \\ \geqslant\ \beta\sum_i\Big(\mu^i-\Gamma_{1-\epsilon}(\mu^i)\Big)-\zeta.\end{aligned}$$

Proof of S:ince $\sum_{j\neq i} G^j(x) = 1 - G^i(x)$, we get

$$\begin{aligned}&\mathbf{E}_{x\sim_{1-\epsilon}y}\Big[\sum_{i,j\in\mathcal{L}} d(i,j)G^i(x)G^j(y)\Big] \\ \geqslant\ &\beta\,\mathbf{E}_{x\sim_{1-\epsilon}y}\Big[\sum_i G^i(x)\big(1-G^i(y)\big)\Big] \\ =\ &\beta\sum_i\Big(\mathbf{E}_x[G^i]-\mathbf{E}_x[G^iT_{1-\epsilon}(G^i)]\Big).\end{aligned}$$

To derive the second part of the lemma, apply Theorem 2.10 on each of the functions G^i with the error term ζ/k instead of ζ:

$$\mathbf{E}_x[G^i]-\mathbf{E}_x[G^iT_{1-\epsilon}(G^i)] \geqslant \Big(\mu^i-\Gamma_{1-\epsilon}(\mu^i)\Big)-\zeta/k.$$

Summing up over all i, we obtain the desired result. □

The following lemma lower bounds the cost of the vertex test, when none of the functions G^i are neither constant nor have an influential variable.

Lemma 3.4 *There exists an ϵ_0 such that for all $\epsilon < \epsilon_0$, for all $\mu = (\mu_1,\mu_2,\ldots,\mu_k) \in \Delta_k$ such that* $\max_i \mu_i < 1-\epsilon^{1/4}$,

$$\sum_i[\mu_i - \Gamma_{1-\epsilon}(\mu_i)] = \Omega(\epsilon^{3/4}).$$

Proof of L:et $\theta = \max_i \mu_i$, then we have $1-\epsilon^{1/4} > \theta$. By choosing $\epsilon < \frac{1}{k^4}$, one can ensure that $\theta > \epsilon^{1/6}$. Observe that Γ satisfies, $\Gamma_{1-\epsilon}(x) \leqslant x$ for all $x \in [0,1]$. Thus we can write,

$$\sum_i[\mu_i - \Gamma_{1-\epsilon}(\mu_i)] > \theta - \Gamma_{1-\epsilon}(\theta).$$

Using known estimates, (see Corollary 10.4 in [18]), we have:

$$\Gamma_{1-\epsilon}(\theta) \leqslant \theta\Big[1-\sqrt{\Omega(\epsilon\log(1/\theta))}\ \Big]+o(\theta).$$

Thus, setting $\theta > \epsilon^{1/4}$, we have the required result:

$$\sum_i[\mu_i - \Gamma_\rho(\mu_i)] \geqslant \Omega(\theta\sqrt{\epsilon\log(1/\theta)}\ ) \geqslant \Omega(\epsilon^{3/4}).$$

□

Lemma 3.5 *For any vertex u,*

$$\mathrm{Val}_{\Lambda}^{\mathsf{Edge}}(u) \geqslant \sum_{e=(a,b)} w_e\, d_{\bowtie}(\mu_{u,a}, \mu_{u,b}).$$

Proof of F:ix an edge $e = (a, b) \in E(\mathcal{H})$. Define

$$f_{ij}^{a,b} = \mathbf{E}_{x\sim_e y}[G_{u,a}^i(x) G_{u,b}^j(y)].$$

Then, $\sum_i f_{ij}^{a,b} = \mu_{u,b}^j$; $\sum_j f_{ij}^{a,b} = \mu_{u,b}^i$. From Definition 2.4 we have

$$\sum_{i,j} d(i,j) f_{ij}^{a,b} \geqslant d_{\bowtie}(\mu_{u,a}, \mu_{u,b}).$$

Recall that $\mathrm{Val}_{\Lambda}^{\mathsf{Edge}}(u)$ is given by

$$\begin{aligned}\mathrm{Val}_{\Lambda}^{\mathsf{Edge}}(u) &= \sum_{e=a,b} w_e \sum_{ij} d(i,j) \mathbf{E}_{x\sim_e y}[G_{u,a}^i(x) G_{u,b}^j(y)] \\ &= \sum_{e=(a,b)} w_e \sum_{ij} d(i,j) f_{ij}^{a,b} \geqslant \sum_{e} w_e\, d_{\bowtie}(\mu_{u,a}, \mu_{u,b}).\end{aligned}$$

□

Lemma 3.6 *There exists $\epsilon_1 > 0$, such that for every $\epsilon < \epsilon_1$, there exist τ, t, such that for all $u \in X$, if $\mathrm{Inf}_j^{<t}(G_{u,a}^i) < \tau$ for all i, j, a, then one of the following inequalities holds:*

$$\mathrm{Val}_{\Lambda}^{\mathsf{Edge}}(u) \geqslant \mathsf{OPT}(\mathcal{H})(1 - 4\epsilon^{1/8})$$

or

$$\mathrm{Val}_{\Lambda}^{\mathsf{Vertex}}(u) \geqslant (\beta\epsilon^{3/4} - \epsilon)/m.$$

Proof of L:et t, τ be as obtained from Lemma 3.3 by setting $\zeta = \epsilon$. Since u is fixed we shall denote $G_{u,a}^i$ by G_a^i. Then, there are two possibilities:

Case 1: For all a, the functions G_a^i are near constant, i.e there is a labeling function $\Lambda : V(\mathcal{H}) \to [k]$ such that $\mu_{u,a}^{\Lambda(a)} > 1 - \epsilon^{1/4}$ for all a.

Set $\theta = \epsilon^{1/8}$. A simple averaging argument shows that for every a, $G_a^{\Lambda(a)}(x) > 1 - \theta$ for a $1 - \theta$ fraction of x. By a union bound, if x, y are generated from $x \sim_e y$, both $G_a^{\Lambda(a)}(x), G_b^{\Lambda(b)}(y)$ are greater than $1 - \theta$ with probability $1 - 2\theta$. Thus the cost of the edge test is:

$$\begin{aligned}\mathrm{Val}_{\Lambda}^{\mathsf{Edge}} &= \mathbf{E}_{(a,b)\in E(\mathcal{H})} \mathbf{E}_{x,y}\Big[\sum_{i,j\in\mathcal{L}} d(i,j) G_{u,a}^i(x) G_{u,b}^j(y)\Big] \\ &\geqslant (1-2\theta)(1-\theta)^2 \mathbf{E}_{(a,b)\in E(\mathcal{H})}\left[d(\Lambda(a), \Lambda(b))\right].\end{aligned}$$

It is easy to check that the labeling Λ is a valid metric labeling solution for $\mathcal{H}$. Hence we have,

$$\mathbf{E}_{(a,b)\in E(\mathcal{H})}\left[d(\Lambda(a), \Lambda(b))\right] \geqslant \mathsf{OPT}(\mathcal{H}).$$

Substituting we get $\mathrm{Val}_{\Lambda}^{\mathsf{Edge}} \geqslant \mathsf{OPT}(\mathcal{H})(1 - 4\theta)$.

Case 2: There exists $b \in V(\mathcal{H})$ such that for all i, $\mu_{u,b}^i \leqslant 1 - \epsilon^{1/4}$. Then, the vertex cost is:

$$
\begin{aligned}
\mathrm{Val}_\Lambda^{\mathsf{Vertex}} &= \mathbf{E}_{a\in V(\mathcal{H})}\mathbf{E}_{x,y}\sum_{i,j\in\mathcal{L}}\left[d(i,j)G^i_{u,a}(x)G^j_{u,a}(y)\right] \\
&\geqslant \frac{1}{m}\mathbf{E}_{x,y}\sum_{i,j\in\mathcal{L}}\left[d(i,j)G^i_{u,b}(x)G^j_{u,b}(y)\right] \\
&\geqslant \frac{1}{m}\left(\beta\sum_i(\mu^i_{u,b}-\Gamma_{1-\epsilon}(\mu^i_{u,b}))-\epsilon\right) \\
&\geqslant \frac{1}{m}\left(\beta\epsilon^{3/4}-\epsilon\right).
\end{aligned}
$$

□

Theorem 3.7 *For every $\gamma>0$, for sufficiently small $\epsilon,\eta>0$, if the UG instance Υ is at most η satisfiable, then the value of the metric labeling instance $(\mathcal{G},\mathcal{L},d)$ obtained from the reduction is at least $\epsilon^{7/8}\,\mathsf{OPT}(\mathcal{H})(1-\gamma)$.*

Proof of S:et $\epsilon\leqslant\min\{(\gamma/12)^8,(\beta/4m\,\mathsf{OPT}(\mathcal{H}))^8,\epsilon_1\}$ and $\delta=\epsilon^{7/8}$. Let t,τ be as obtained from Lemma 3.6. For every vertex $u\in\mathcal{X}$, one of the following is true:

- There exists $a\in V(\mathcal{H})$, $i\in[k]$, $j\in[R]$ such that $\mathrm{Inf}_j^{<t}(G^i)\geqslant\tau$.
- $\mathrm{Val}_\Lambda^{\mathsf{Edge}}(u)\geqslant\mathsf{OPT}(\mathcal{H})(1-4\epsilon^{1/8})\geqslant\mathsf{OPT}(\mathcal{H})(1-\gamma/3)$.
- $\mathrm{Val}_\Lambda^{\mathsf{Vertex}}(u)\geqslant(\beta\epsilon^{3/4}-\epsilon)/m\geqslant\delta\,\mathsf{OPT}(\mathcal{H})$.

Thus, if all the functions associated with a particular $u\in X$ have low influence,

$$\delta\,\mathrm{Val}_\Lambda^{\mathsf{Edge}}(u)+(1-\delta)\,\mathrm{Val}_\Lambda^{\mathsf{Vertex}}(u)\geqslant\delta\,\mathsf{OPT}(\mathcal{H})(1-\gamma/3).$$

Call a vertex $u\in\mathcal{X}$ *good* if at least one of the functions associated with it has an influential variable. More precisely, if there exists a,i,j such that $\mathrm{Inf}_j^{<t}(G^i_{u,a})\geqslant\tau$. If $\mathrm{Val}_\Lambda(\mathcal{G})\leqslant\delta\,\mathsf{OPT}(\mathcal{H})(1-\gamma)$, then at least $\gamma/2$ fraction of the vertices are *good*. Fix a *good* vertex $u\in\mathcal{X}$ with the corresponding a,i,j satisfying $\mathrm{Inf}_j^{<t}(G^i_{u,a})\geqslant\tau$. Then,

$$
\begin{aligned}
\tau &\leqslant \sum_{\sigma_j\neq0,|\sigma|\leqslant t}|\widehat{G^i_{u,a}}(\sigma)|^2 \\
&\leqslant \sum_{\sigma_j\neq0,|\sigma|\leqslant t}\mathbf{E}_w|\widehat{F^i_{w,a}}(\pi^{-1}_{u,w}(\sigma))|^2 \\
&=\mathbf{E}_w\left[\mathrm{Inf}^{<t}_{\pi^{-1}_{u,w}(j)}(F^i_{w,a})\right].
\end{aligned}
$$

We will define the "decoding" for vertices of the UG instance as follows:

$$
\begin{aligned}
\Lambda(u) &= \{j\in[R]\ \mid\ \exists\, i,a;\ \mathrm{Inf}_j^{<t}(G^i_{u,a})\geqslant\tau\} \\
&\qquad\qquad\qquad\text{(for every } u\in\mathcal{X}\text{)}, \\
\Lambda(w) &= \{j\in[R]\ \mid\ \exists\, i,a;\ \mathrm{Inf}_j^{<t}(F^i_{w,a})\geqslant\tau/2\} \\
&\qquad\qquad\qquad\text{(for every } w\in\mathcal{Y}\text{)}.
\end{aligned}
$$

Using Fact 2.8, the sizes of the sets above are at most $2tkm/\tau$. We will analyze the fraction of edges in the UG instance satisfied when we assign a label uniformly at random from $\Lambda(u)$ independently for every $u\in\mathcal{X}\cup\mathcal{Y}$. For a good vertex u, for every $r\in\Lambda(u)$, the fraction of neighbors that have a satisfying label in its decoding is at least $\tau/2$. Moreover such an edge is satisfied with probability at least $\tau^2/4t^2k^2m^2$. Thus, the weight of edges satisfied is at least $(\gamma/2)(\tau/2)(\tau^2/4t^2k^2m^2)\geqslant\frac{\gamma\tau^3}{16t^2m^2k^2}$. Choosing $\eta<\frac{\gamma\tau^3}{16t^2m^2k^2}$ gives the required result. □

4 Integrality Gap for SDPs

In this section, we construct integrality gaps for the semidefinite relaxation (EM-SDP) using the unique games hardness reduction. In particular, this will yield SDP integrality gaps that match the integrality gap of the earthmover LP relaxation (EM-LP). Our starting point is the Unique Games integrality gap instance constructed by Khot and Vishnoi [21].

First, the integrality gap instance $I = (V, E)$ presented in [21] is not bipartite. To obtain a bipartite unique games instance I', duplicate the vertices by setting $\mathcal{X} = \{(v,0) | v \in V\}$ and $\mathcal{Y} = \{(v,1) | v \in V\}$. Further for each edge $(u,v) \in E$, introduce two edges $((u,0),(v,1))$ and $((u,1),(v,0))$ in I'. The SDP solution for the bipartite instance I' is obtained by assigning the vector corresponding to $v \in V$ to both vertices $(v,0)$ and $(v,1)$. Except for these minor modifications, the following theorem is a direct consequence of [21]

Theorem 4.1 *For every $\eta > 0$, there exists a UG instance, $\Upsilon = (\mathcal{X} \cup \mathcal{Y}, E, \Pi = \{\pi_e : [R] \to [R] \mid e \in E\}, [R])$ and vectors $\{u_w^i\}$ for every $w \in \mathcal{Y}$, $i \in [R]$ such that the following conditions hold :*

- *No assignment satisfies more than η fraction of constraints in Υ.*
- *For all $w, w_1, w_2 \in \mathcal{Y}, i, j \in [R]$,*

$$u_{w_1}^i \cdot u_{w_2}^j \geqslant 0 \quad \textit{and} \quad u_w^i \cdot u_w^j = 0.$$

- *For all $w, w_1, w_2 \in \mathcal{Y}, i, j \in [R]$,*

$$u_{w_1}^i \cdot \sum_{j \in [R]} u_{w_2}^j = |u_{w_1}^i|^2 \ \textit{and} \ \sum_{i \in [R]} |u_w^i|^2 = 1.$$

- *The SDP value is at least $1 - \eta$:*

$$E_{v \in \mathcal{X}, w_1, w_2 \in \mathcal{Y}} \Big[\sum_{r \in R} u_{w_1}^{\pi_1(i)} \cdot u_{w_2}^{\pi_2(i)} \Big] \geqslant 1 - \eta. \tag{$*$}$$

Given a (structured) integrality gap instance for the linear programming relaxation (EM-LP), we apply the reduction in Section 3 to a unique games instance provided by the above theorem. Applying Theorem 3.7 directly gives the lower bound on the integral optimum of the instance.

We obtain the SDP vectors for a vertex $z = (w, a, x) \in V(\mathcal{G})$ as follows: Intuitively, if the vertex had label j, we would assign z the label x_j. For every vertex $z = (w, a, x) \in V(\mathcal{G})$, we construct SDP vectors as follows: Partition $[R]$ into k parts, $P_z(i)$, $i \in \mathcal{L}$:

$$P_z(i) = \{j \mid x_j = i\}.$$

The vector for label i is then the sum of the vectors in the partition $P_z(i)$:

$$v_z^i = \sum_{j \in P_v(i)} u_w^j \quad \forall z = (w, a, x).$$

The SDP optimum can be bounded in terms of $(*)$ in Theorem 4.1. We state it formally in the following theorem.

Theorem 4.2 *For every structured integrality gap instance, $\Psi = (\mathcal{H}, \mathcal{L}, d, \{X_v\})$ and sufficiently small γ, ϵ, η if Υ denotes the UG instance obtained from Theorem 4.1, then the metric labeling instance $(\mathcal{G}, \mathcal{L}, d)$ obtained using the reduction has the following properties:*

$$\mathsf{OPT}(\mathcal{G}) \geqslant \epsilon^{7/8}\,\mathsf{OPT}(\mathcal{H})(1 - \gamma)$$
$$\mathsf{SDP}(\mathcal{G}) \leqslant \epsilon^{7/8}\,\mathsf{LP}(\mathcal{H})(1 + \gamma).$$

Proof of T:he UG instance Υ obtained from Theorem 4.1 is at most η satisfiable. Hence, Theorem 3.7 implies that the value $\mathcal{G}$, $\mathsf{OPT}(\mathcal{G})$ is at least $\epsilon^{7/8}\,\mathsf{OPT}(\mathcal{H})(1-\gamma)$ for η, ϵ sufficiently small.

For every vertex $z = (w, a, x) \in V(\mathcal{G})$, we construct SDP vectors as follows: Partition $[R]$ into k parts, $P_z(i)$, $i \in \mathcal{L}$:

$$P_z(i) = \{j \mid x_j = i\}.$$

The vector for label i is then the sum of the vectors in the partition $P_z(i)$:

$$v_z^i = \sum_{j \in P_v(i)} u_w^j \quad \forall z = (w, a, x).$$

It is easy to see that constraints (1), (2), (3), (4) and (5) of the SDP (EM-SDP) are satisfied by the vectors v_z^i. Set $\delta = \epsilon^{7/8}$. Now, the SDP cost of the vectors can be bounded as follows:

$$\begin{aligned}
\mathsf{SDP}(\mathcal{G}) = \mathbf{E}_{v,w_1,w_2}\Bigg[&\delta \cdot \mathbf{E}_{(a,b)\in E(\mathcal{H})} \\
&\mathbf{E}_{x \sim_\epsilon y} \sum_{i,j} d(i,j) v^i_{w_1,a,(\pi_1(x))} \cdot v^j_{w_2,b,(\pi_2(y))} \\
&+ (1-\delta)\cdot \mathbf{E}_{a\in V(\mathcal{H})} \\
&\mathbf{E}_{x,y} \sum_{i,j} d(i,j) v^i_{w_1,a,(\pi_1(x))} \cdot v^j_{w_2,a,(\pi_2(y))}\Bigg] \\
= \mathbf{E}_{v,w_1,w_2}\Bigg[&\delta \cdot \mathbf{E}_{(a,b)\in E(\mathcal{H})} \Big\{ \sum_{i,j} d(i,j) \sum_{r_1,r_2} u_{w_1}^{\pi_1(r_1)} \\
&\cdot u_{w_2}^{\pi_2(r_2)} \Pr[x_{r_1} = i \wedge y_{r_2} = j\Big\} + (1-\delta)\cdot \mathbf{E}_{a\in V(\mathcal{H})} \\
&\Big\{ \sum_{i,j} d(i,j) \sum_{r_1,r_2} u_{w_1}^{\pi_1(r_1)} \\
&\cdot u_{w_2}^{\pi_2(r_2)} \Pr[x_{r_1} = i \wedge y_{r_2} = j]\Big\}\Bigg].
\end{aligned}$$

Additionally, the last expression is bounded from above by:

$$\begin{aligned}
&\leqslant \mathbf{E}_{v,w_1,w_2}\Bigg[\delta \cdot \Big\{ \mathsf{LP}(\mathcal{H}) \sum_r u_{w_1}^{\pi_1(r_1)} \cdot u_{w_2}^{\pi_2(r_2)} \\
&\quad + \sum_{r_1 \neq r_2} u_{w_1}^{\pi_1(r_1)} \cdot u_{w_2}^{\pi_2(r_2)} \Big\} + (1-\delta) \\
&\quad \cdot \Big\{ \epsilon \sum_r u_{w_1}^{\pi_1(r_1)} \cdot u_{w_2}^{\pi_2(r_2)} + \sum_{r_1 \neq r_2} u^i_{w_1,r_1} \cdot u^j_{w_2,r_2} \Big\}\Bigg] \\
&\leqslant (\delta\,\mathsf{LP}(\mathcal{H}) + \epsilon) \cdot \mathbf{E}_{v,w_1,w_2}\left[\sum_r u_{w_1}^{\pi_1(r_1)} \cdot u_{w_2}^{\pi_2(r_2)}\right] \\
&\quad + 1 - \mathbf{E}_{v,w_1,w_2}\left[\sum_r u_{w_1}^{\pi_1(r_1)} \cdot u_{w_2}^{\pi_2(r_2)}\right] \\
&\leqslant (1-\eta)\cdot(\delta\,\mathsf{LP}(\mathcal{H}) + \epsilon) + \eta.
\end{aligned}$$

From Corollary 3.2, we have that for sufficiently small η, ϵ:

$$\mathsf{SDP}(\mathcal{G}) \leqslant \epsilon^{7/8}\,\mathsf{LP}(\mathcal{H})(1+\gamma).$$

$\square$

5 Acknowledgments

We would like to thank Sanjeev Arora, Venkatesan Guruswami, and Kunal Talwar for many fruitful discussions. We would also like to thank the anonymous reviewers for their valuable suggestions to improve the presentation

References

[1] Aaron Archer, Jittat Fakcharoenphol, Chris Harrelson, Robert Krauthgamer, Kunal Talwar, and Éva Tardos. Approximate classification via earthmover metrics. In *SODA '04: Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1079–1087, 2004.

[2] Per Austrin. Balanced max 2-sat might not be the hardest. In *STOC '07: Proceedings of the 39th Annual ACM Symposium on Theory of Computing*, pages 189–197, 2007.

[3] Per Austrin. Towards sharp inapproximability for any 2-csp. In *FOCS '07: Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, pages 307–317, 2007.

[4] Yair Bartal. On approximating arbitrary metrices by tree metrics. In *STOC '98: Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pages 161–168, 1998.

[5] Gruia Calinescu, Howard Karloff, and Yuval Rabani. Approximation algorithms for the 0-extension problem. In *SODA '01: Proceedings of the 12th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 8–16, 2001.

[6] Chandra Chekuri, Sanjeev Khanna, Joseph (Seffi) Naor, and Leonid Zosin. Approximation algorithms for the metric labeling problem via a new linear programming formulation. *SIAM J. Disc. Math.*, 18:608–625, 2005.

[7] Julia Chuzhoy and Joseph (Seffi) Naor. The hardness of metric labeling. *SIAM J. Comput.*, 36(5):1376–1386, 2006.

[8] Gruia Călinescu, Howard Karloff, and Yuval Rabani. An improved approximation algorithm for multiway cut. In *STOC '98: Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pages 48–52, 1998.

[9] Elias Dahlhaus, David Johnson, Christos Papadimitriou, Paul Seymour, and Mihalis Yannakakis. The complexity of multiway cuts (extended abstract). In *STOC '92: Proceedings of the 24th Annual ACM Symposium on Theory of Computing*, pages 241–251, 1992.

[10] Jittat Fakcharoenphol, Chris Harrelson, Satish Rao, and Kunal Talwar. An improved approximation algorithm for the 0-extension problem. In *SODA '03: Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 257–265, 2003.

[11] Jittat Fakcharoenphol, Satish Rao, and Kunal Talwar. Approximating metrics by tree metrics. *SIGACT News*, 35(2):60–70, 2004.

[12] Anupam Gupta and Éva Tardos. A constant factor approximation algorithm for a class of classification problems. In *STOC '00: Proceedings of the 32nd Annual ACM Symposium on Theory of Computing*, pages 652–658, 2000.

[13] David R. Karger, Philip Klein, Cliff Stein, Mikkel Thorup, and Neal E. Young. Rounding algorithms for a geometric embedding of minimum multiway cut. In *STOC '99: Proceedings of the 31st Annual ACM Symposium on Theory of Computing*, pages 668–678, 1999.

[14] Howard Karloff, Subhash Khot, Aranyak Mehta, and Yuval Rabani. On earthmover distance, metric labeling, and 0-extension. In *STOC '06: Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, pages 547–556, 2006.

[15] Alexander Karzanov. Minimum 0-extension of graph metrics. *Europ. J. Combinat.*, 19:71–101, 1998.

[16] Alexander Karzanov. A combinatorial algorithm for the minimum $(2,r)$-metric problem and some generalizations. *Combinatorica*, 18:549–569, 1999.

[17] Subhash Khot. On the power of unique 2-prover 1-round games. In *STOC '02: Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, pages 767–775, 2002.

[18] Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan O'Donnell. Optimal inapproximability results for max-cut and other 2-variable csps? In *FOCS '04: Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 146–154, 2004.

[19] Subhash Khot and Ryan O'Donnell. SDP gaps and UGC-hardness for maxcutgain. In *FOCS '06: Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science*, pages 217–226, 2006.

[20] Subhash Khot and Oded Regev. Vertex cover might be hard to approximate to within 2-ϵ. *J. Comput. Syst. Sci.*, 74(3):335–349, 2008.

[21] Subhash A. Khot and Nisheeth K. Vishnoi. The unique games conjecture, integrality gap for cut problems and embeddability of negative type metrics into ℓ_1. In *FOCS '05: Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, pages 53–62, 2005.

[22] Jon Kleinberg and Éva Tardos. Approximation algorithms for classification problems with pairwise relationships: metric labeling and markov random fields. *J. ACM*, 49(5):616–639, 2002.

[23] Robert Krauthgamer, James R. Lee, Manor Mendel, and Assaf Naor. Measured descent: A new embedding method for finite metrics. In *FOCS '04: Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 434–443, 2004.

[24] James R. Lee and Assaf Naor. Extending lipschitz functions via random metric partitions. *Inventiones Mathematicae*, 160(1):59–95, 2005.

[25] Elchanan Mossel, Ryan O'Donnell, and Krzysztof Oleszkiewicz. Noise stability of functions with low influences: Invariance and optimality. *In FOCS '05: Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, pages 21–30, 2005.

[26] Prasad Raghavendra. Optimal algorithms and inapproximability results for every csp? *In STOC '08: Proceedings of the 40th ACM Symposium on Theory of Computing*, 2008.