Expander Flows, Geometric Embeddings and Graph Partitioning

SANJEEV ARORA

Princeton University, Princeton, New Jersey

AND

SATISH RAO AND UMESH VAZIRANI

UC Berkeley, Berkeley, California

Abstract. We give a $O(\sqrt{\log n})$-approximation algorithm for the SPARSEST CUT, EDGE EXPANSION, BALANCED SEPARATOR, and GRAPH CONDUCTANCE problems. This improves the $O(\log n)$-approximation of Leighton and Rao (1988). We use a well-known semidefinite relaxation with triangle inequality constraints. Central to our analysis is a geometric theorem about projections of point sets in $\Re^d$, whose proof makes essential use of a phenomenon called measure concentration.

We also describe an interesting and natural "approximate certificate" for a graph's expansion, which involves embedding an n-node expander in it with appropriate dilation and congestion. We call this an expander flow.

Categories and Subject Descriptors: F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems; G.2.2 [**Discrete Mathematics**]: Graph Theory

General Terms: Algorithms, Theory

Additional Key Words and Phrases: Graph partitioning, semidefinite programs, graph separators, multicommodity flows, expansion, expanders

1. *Introduction*

Partitioning a graph into two (or more) large pieces while minimizing the size of the "interface" between them is a fundamental combinatorial problem. Graph partitions or separators are central objects of study in the theory of Markov chains, geometric embeddings and are a natural algorithmic primitive in numerous settings, including clustering, divide and conquer approaches, PRAM emulation, VLSI layout, and packet routing in distributed networks. Since finding optimal separators is **NP**-hard, one is forced to settle for approximation algorithms (see Shmoys [1995]). Here we give new approximation algorithms for some of the important problems in this class.

Graph partitioning involves finding a cut with few crossing edges conditioned on or normalized by the size of the smaller side of the cut. The problem can be made precise in different ways, giving rise to several related measures of the quality of the cut, depending on precisely how size is measured, such as conductance, expansion, normalized or sparsest cut. Precise definitions appear in Section 2. These measures are approximation reducible within a constact factor Leighton and Rao [1999], are all **NP**-hard, and arise naturally in different contexts.

A weak approximation for GRAPH CONDUCTANCE follows from the connection—first discovered in the context of Riemannian manifolds Cheeger [1970] —between conductance and the eigenvalue gap of the Laplacian: $2\Phi(G) \geq \lambda \geq \Phi(G)^2/2$ [Alon and Milman 1985; Alon 1986; Jerrum and Sinclair 1989]. The approximation factor is $1/\Phi(G)$, and hence $\Omega(n)$ in the worst case, and $O(1)$ only if $\Phi(G)$ is a constant. This connection between eigenvalues and expansion has had enormous influence in a variety of fields (see, e.g., Chung [1997]).

Leighton and Rao [1999] designed the first true approximation by giving $O(\log n)$-approximations for SPARSEST CUT and GRAPH CONDUCTANCE and $O(\log n)$-pseudo-approximations[1] for c-BALANCED SEPARATOR. They used a linear programming relaxation of the problem based on multicommodity flow proposed in Shahrokhi and Matula [1990]. These ideas led to approximation algorithms for numerous other **NP**-hard problems, see Shmoys [1995]. We note that the integrality gap of the LP is $\Omega(\log n)$, and therefore improving the approximation factor necessitates new techniques.

In this article, we give $O(\sqrt{\log n})$-approximations for SPARSEST CUT, EDGE EXPANSION, and GRAPH CONDUCTANCE and $O(\sqrt{\log n})$-pseudo-approximation to c-BALANCED SEPARATOR. As we describe below, our techniques have also led to new approximation algorithms for several other problems, as well as a breakthrough in geometric embeddings.

The key idea underlying algorithms for graph partitioning is to spread out the vertices in some abstract space while not stretching the edges too much. Finding a

---

[1] For any fixed $c' < c$ the pseudo-approximation algorithm finds a c'-balanced cut whose expansion is at most $O(\log n)$ times expansion of the best c-balanced cut.

good graph partition is then accomplished by partitioning this abstract space. In the eigenvalue approach, the vertices are mapped to points on the real line such that the average squared distance is constant, while the average squared distance between the endpoints of edges is minimized. Intuitively, snipping this line at a random point should cut few edges, thus yielding a good cut. In the linear programming approach, lengths are assigned to edges such that the average distance between pairs of vertices is fixed, while the average edge length is small. In this case, it can be shown that a ball whose radius is drawn from a particular distribution defines a relatively balanced cut with few expected crossing edges, thus yielding a sparse cut.

Note that by definition, the distances in the linear programming approach form a metric (i.e., satisfy the triangle inequality) while in the eigenvalue approach they don't. On the other hand, the latter approach works with a geometric embedding of the graph, whereas there isn't such an underlying geometry in the former.

In this article, we work with an approach that combines both geometry and the metric condition. Consider mapping the vertices to points on the unit sphere in $\Re^n$ such that the squared distances form a metric. We refer to this as an ℓ_2^2-*representation* of the graph. We say that it is *well-spread* if the average squared distance among all vertex pairs is a fixed constant, say ρ. Define the *value* of such a representation to be the sum of squared distances between endpoints of edges. The value of the *best* representation is (up to a factor 4) a lower bound on the capacity of the best c-balanced separator, where $4c(1 - c) = \rho$. The reason is that every c-balanced cut in the graph corresponds to a ℓ_2^2 representation in a natural way: map each side of the cut to one of two antipodal points. The value of such a representation is clearly 4 times the cut capacity since the only edges that contribute to the value are those that cross the cut, and each of them contributes 4. The average squared distance among pairs of vertices is at least $4c(1 - c)$.

Our approach starts by finding a well-spread representation of minimum value, which is possible in polynomial time using semidefinite programming. Of course, this minimum value representation will not in general correspond to a cut. The crux of this paper is to extract a low-capacity balanced cut from this embedding.

The key to this is a new result (Theorem 1) about the geometric structure of well-spread ℓ_2^2-representations: they contain $\Omega(n)$ sized sets S and T that are well-separated, in the sense that every pair of points $v_i \in S$ and $v_j \in T$ must be at least $\Delta = \Omega(1/\sqrt{\log n})$ apart in ℓ_2^2 (squared Euclidean) distance. The set S can be used to find a good cut as follows: consider all points within some distance $\delta \in [0, \Delta]$ from S, where δ is chosen uniformly at random. The quality of this cut depends upon the value of representation. In particular, if we start with a representation of minimum value, the expected number of edges crossing such a cut must be small, since the length of a typical edge is short relative to Δ.

Furthermore, the sets S and T can be constructed algorithmically by projecting the points on a random line as follows (Section 3): for suitable constant c, the leftmost cn and rightmost cn points on the line are our first candidates for S and T. However, they can contain pairs of points $v_i \in S$, $v_j \in T$ whose squared distance is less than Δ, which we discard. The technically hard part in the analysis is to prove that not too many points get discarded. This delicate argument makes essential use of a phenomenon called *measure concentration*, a cornerstone of modern convex geometry (see Ball [1997]).

The result on well-separated sets is tight for an n-vertex hypercube—specifically, its natural embedding on the unit sphere in $\Re^{\log n}$ which defines an ℓ_2^2 metric—where measure concentration implies that any two large sets are within $O(1/\sqrt{\log n})$ distance. The hypercube example appears to be a serious obstacle to improving the approximation factor for finding sparse cuts, since the optimal cuts of this graph (namely, the dimension cuts) elude all our rounding techniques. In fact, hypercube-like examples have been used to show that this particular SDP approach has an integrality gap of $\Omega(\log \log n)$ [Devanuretal et al. 2006; Krauthgamer and Rabani 2006].

Can our approximation ratio be improved over $O(\sqrt{\log n})$? In Section 8, we formulate a sequence of conditions that would imply improved approximations. Our theorem on well-separated sets is also important in study of metric embeddings, which is further discussed below.

1.1. EXPANDER FLOWS.   While SDPs can be solved in polynomial time, the running times in practice are not very good. Ideally, one desires a faster algorithm, preferably one using simple primitives like shortest paths or single commodity flow. In this article, we initiate such a combinatorial approach that realizes the same approximation bounds as our SDP-based algorithms, but does not use SDPs. Though the resulting running times in this article are actually inferior to the SDP-based approach, subsequent work has led to algorithms that are significantly faster than the SDP-based approach.

We design our combinatorial algorithm using the notion of *expander flows*, which constitute an interesting and natural "certificate" of a graph's expansion. Note that any algorithm that approximates edge expansion $\alpha = \alpha(G)$ must implicitly certify that every cut has large expansion. One way to do this certification is to embed[2] a complete graph into the given graph with minimum congestion, say μ. (Determining μ is a polynomial-time computation using linear programming.) Then, it follows that every cut must have expansion at least n/μ. (See Section 7.) This is exactly the certificate used in the Leighton-Rao paper, where it is shown that congestion $O(n \log n/\alpha(G))$ suffices (and this amount of congestion is required on some worst-case graphs). Thus, embedding the complete graph suffices to certify that the expansion is at least $\alpha(G)/\log n$.

Our certificate can be seen as a generalization of this approach, whereby we embed not the complete graph but some flow that is an "expander" (a graph whose edge expansion is $\Omega(1)$). We show that for every graph there is an expander flow that certifies an expansion of $\Omega(\alpha(G)/\sqrt{\log n})$ (see Section 7). This near-optimal embedding of an expander inside an arbitrary graph may be viewed as a purely structural result in graph theory. It is therefore interesting that this graph-theoretic result is proved using geometric arguments similar to the ones used to analyse our SDP-based algorithm. (There is a natural connection between the two algorithms since expander flows can be viewed as a natural family of dual solutions to the above-mentioned SDP, see Section 7.4.) In fact, the expander flows approach was the original starting point of our work.

---

[2] Note that this notion of graph embedding has no connection in general to geometric embeddings of metric spaces, or geometric embeddings of graphs using semidefinite programming. It is somewhat confusing that all these notions of embeddings end up being relevant in this article.

How does one efficiently compute an expander flow? The condition that the multicommodity flow is an expander can be imposed by exponentially many linear constraints. For each cut, we have a constraint stating that the amount of flow across that cut is proportional to the number of vertices on the smaller side of the cut. We use the Ellipsoid method to find the feasible flow. At each step, we have to check if any of these exponentially many constraints are violated. Using the fact that the eigenvalue approach gives a constant factor approximation when the edge expansion is large, we can efficiently find violated constraints (to within constant approximation), and thus find an approximately feasible flow in polynomial time using the Ellipsoid method. Arora et al. [2004] used this framework to design an efficient $\tilde{O}(n^2)$ time approximation algorithm, thus matching the running time of the best implementations of the Leighton–Rao algorithm. Their algorithm may be thought of as a primal-dual algorithm and proceeds by alternately computing eigenvalues and solving min-cost multicommodity flows. Khandekar et al. [2006] use the expander flow approach to break the $\tilde{O}(n^2)$ multicommodity flow barrier and rigorously analyse an algorithm that resembles some practially effective heuristics [Lang and Rao 2004]. The analysis gives a worse $O(\log^2 n)$ approximation for SPARSEST CUT but requires only $O(\log^2 n)$ invocations of a single commodity maximum flow algorithm. Arora and Kale [2007] have improved this approximation factor to $O(\log n)$ with a different algorithm. Thus one has the counterintuitive result that an SDP-inspired algorithm runs faster than the older LP-inspired algorithm a la Leighton-Rao, while attaining the same approximation ratio. Arora and Kale also initiate a new primal-dual and combinatorial approach to other SDP-based approximation algorithms.

1.2. Related Prior and Subsequent Work.

1.2.1. *Semidefinite Programming and Approximation Algorithms.* Semidefinite programs (SDPs) have numerous applications in optimization. They are solvable in polynomial time via the ellipsoid method Grötschel et al. [1993], and more efficient interior point methods are now known [Alizadeh 1995; Nesterov and Nemirovskii 1994]. In a seminal paper, Goemans and Williamson [1995] used SDPs to design good approximation algorithms for MAX-CUT and MAX-k-SAT. Researchers soon extended their techniques to other problems [Karger et al. 1998; Karloff and Zwick 1997; Goemans 1998], but lately progress in this direction had stalled. Especially in the context of minimization problems, the GW approach of analyzing "random hyperplane" rounding in an edge-by-edge fashion runs into well-known problems. By contrast, our theorem about well separated sets in ℓ_2^2 spaces (and the "rounding" technique that follows from it) takes a more global view of the metric space. It is the mathematical heart of our technique, just as the *region-growing* argument was the heart of the Leighton-Rao technique for analyzing LPs.

Several papers have pushed this technique further, and developed closer analogs of the region-growing argument for SDPs using our main theorem. Using this they design new $\sqrt{\log n}$-approximation algorithms for graph deletion problems such as 2CNF-DELETION and MIN-UNCUT [Agarwal et al. 2005], for MIN-LINEAR ARRANGEMENT [Charikar et al. 2006; Feige and Lee 2007], and VERTEX SEPARATORS [Feige et al. 2005]. Similarly, MIN-VERTEX COVER can now be approximated upto a factor $2 - \Omega(1/\sqrt{\log n})$ [karakostas 2005], an improvement over $2 - \Omega(\log \log n / \log n)$

achieved by prior algorithms. Also, a $O(\sqrt{\log n})$ approximation algorithm for node cuts is given in Feige et al. [2005].

However, we note that the Structure Theorem does not provide a blanket improvement for all the problems for which $O(\log n)$ algorithms were earlier designed using the Leighton-Rao technique. In particular, the integrality gap of the SDP relaxation for MIN-MULTICUT was shown to be $\Omega(\log n)$ [Aggarwal et al. 2005], the same (upto a constant factor) as the integrality gap for the LP relaxation.

1.2.2. *Analysis of Random Walks.* The mixing time of a random walk on a graph is related to the first nonzero eigenvalue of the Laplacian, and hence to the edge expansion. Of various techniques known for upper bounding the mixing time, most rely on lower bounding the conductance. Diaconis and Saloff-Coste [1993] describe a very general idea called the *comparison technique*, whereby the edge expansion of a graph is lower bounded by embedding a *known* graph with *known* edge expansion into it. (The embedding need not be efficiently constructible; existence suffices.) Sinclair [1992] suggested a similar technique and also noted that the Leighton–Rao multicommodity flow can be viewed as a generalization of the Jerrum–Sinclair [1989] canonical path argument. Our results on expander flows imply that the comparison technique can be used to always get to within $O(\sqrt{\log n})$ of the correct value of edge expansion.

1.2.3. *Metric Spaces and Relaxations of the Cut Cone.* The graph partitioning problem is closely related to geometric embeddings of metric spaces, as first pointed out by Linial et al. [1995] and Aumann and Rabani [1998]. The *cut cone* is the cone of all cut semi-metrics, and is equivalent to the cone of all ℓ_1 semi-metrics. Graph separation problems can often be viewed as the optimization of a linear function over the cut cone (possibly with some additional constraints imposed). Thus, optimization over the cut cone is **NP**-hard. However, one could relax the problem and optimize over some other metric space, embed this metric space in ℓ_1 (hopefully, with low distortion), and then derive an approximation algorithm. This approach is surveyed in Shmoys's survey [1995]. The integrality gap of the obvious generalization of our SDP for NONUNIFORM SPARSEST CUT is *exactly* the same as as the worst-case distortion incurred in embedding n-point ℓ_2^2 metrics into ℓ_1. A famous result of Bourgain shows that the distortion is at most $O(\log n)$, which yields an $O(\log n)$-approximation for NONUNIFORM SPARSEST CUT. Improving this has been a major open problem.

The above connection shows that an $\alpha(n)$ approximation for conductance and other measures for graph partitioning would follow from any general upper bound $\alpha(n)$ for the minimum distortion for embedding ℓ_2^2 metrics into ℓ_1. Our result on well-separated sets has recently been used [Arora et al. 2008] to bound this distortion by $O(\sqrt{\log n} \log \log n)$, improving on Bourgain's classical $O(\log n)$ bound that applies to any metric. In fact, they give a stronger result: an embedding of ℓ_2^2 metrics into ℓ_2. Since any ℓ_2 metric is an ℓ_1 metric, which in turn is an ℓ_2^2 metric, this also gives an embedding of ℓ_1 into ℓ_2 with the same distortion. This almost resolves a central problem in the area of metric embeddings, since any ℓ_2 embedding of the hypercube, regarded as an ℓ_1 metric, must suffer an $\Omega(\sqrt{\log n})$ distortion [Enflo 1969].

A key idea in these new developments is our above-mentioned main theorem about the existence of well-separated sets S, T in well-spread ℓ_2^2 metrics. This can

be viewed as a weak "average case" embedding from ℓ_2^2 into ℓ_1 whose distortion is $O(\sqrt{\log n})$. This is pointed out by Chawla et al. [2005], who used it together with the measured descent technique of Krauthgamer et al. [2005] to show that n-point ℓ_2^2 metrics embed into ℓ_2 (and hence into ℓ_1) with distortion $O(\log^{3/4} n)$. Arora et al. [2008] improved this distortion to $O(\sqrt{\log n} \log \log n)$, which is optimal up to $\log \log n$ factor.

To see why our main theorem is relevant in embeddings, realize that usually the challenge in producing such an embedding is to ensure that the images of any two points from the original ℓ_2^2 metric are far enough apart in the resulting ℓ_1 metric. This is accomplished in Arora et al. [2008] by considering dividing possible distances into geometrically increasing intervals and generating coordinates at each distance scale. Our main theorem is used to find a well-separated pair of sets, S and T, for each scale, and then to use "distance to S" as the value of the corresponding coordinate. Since each node in T is at least $\Delta/\sqrt{\log n}$ away from every node in S, this ensures that a large number of Δ-separated pairs of nodes are far enough apart. The $O(\sqrt{\log n} \log \log n)$ bound on the distortion requires a clever accounting scheme across distance scales.

We note that above developments showed that the integrality gap for NONUNIFORM SPARSEST CUT is $O(\sqrt{\log n} \log \log n)$. There was speculation that the integrality gap may even be $O(1)$, but Khot and Vishnoi [2005] have shown it is at least $(\log \log n)^\epsilon$ for some $\epsilon > 0$.

2. *Definitions and Results*

We first define the problems considered in this article; all are easily proved to be **NP**-hard [Leighton and Rao 1999]. Given a graph $G = (V, E)$, the goal in the UNIFORM SPARSEST CUT problem is to determine the cut $(S, \overline{S})$ (where $|S| \leq |\overline{S}|$ without loss of generality) that minimizes

$$\frac{\left|E(S, \overline{S})\right|}{|S|\,|\overline{S}|}. \tag{1}$$

Since $|V|/2 \leq |\overline{S}| \leq |V|$, up to a factor 2 computing the sparsest cut is the same as computing the *edge expansion* of the graph, namely,

$$\alpha(G) = \min_{S \subseteq V, |S| \leq |V|/2} \frac{\left|E(S, \overline{S})\right|}{|S|}.$$

Since factors of 2 are inconsequential in this article, we use the two problems interchangeably. Furthermore, we often shorten UNIFORM SPARSEST CUT to just SPARSEST CUT. In another related problem, c-BALANCED-SEPARATOR, the goal is to determine $\alpha_c(G)$, the minimum edge expansion among all c-balanced cuts. (A cut $(S, \overline{S})$ is *c-balanced* if both $S, \overline{S}$ have at least $c\,|V|$ vertices.) In the GRAPH CONDUCTANCE problem, we wish to determine

$$\Phi(G) = \min_{S \subseteq V, |E(S)| \leq |E|/2} \frac{\left|E(S, \overline{S})\right|}{|E(S)|}, \tag{2}$$

where $E(S)$ denotes the multiset of edges incident to nodes in S (i.e., edges with both endpoints in S are included twice).

There are well-known interreducibilities among these problems for approximation. The edge expansion, α, of a graph is between $n/2$ and n times the optimal sparse cut value. The conductance problem can be reduced to the sparsest cut problem by replacing each node with a degree d with a clique on d nodes; any sparse cut in the transformed graph does not split cliques, and the sparsity of any cut (that does not split cliques) in the transformed graph is within a constant factor of the conductance of the cut. The sparsest cut problem can be reduced to the conductance problem by replacing each node with a large (say sized $C = n^2$ size) clique. Again, small cuts don't split cliques and a (clique nonsplitting) cut of conductance Φ in the transformed graph corresponds to the a cut in the original of sparsity Φ/C^2. The reductions can be bounded degree as well [Leighton and Rao 1999].

As mentioned, all our algorithms depend upon a geometric representation of the graph.

Definition 1 (ℓ_2^2-*representation*).   An ℓ_2^2-*representation* of a graph is an assignment of a point (vector) to each node, say v_i assigned to node i, such that for all i, j, k:

$$\left|v_i - v_j\right|^2 + \left|v_j - v_k\right|^2 \geq |v_i - v_k|^2 \quad \text{(triangle inequality)} \tag{3}$$

An ℓ_2^2-representation is called a *unit-ℓ_2^2 representation* if all points lie on the unit sphere (or equivalently, all vectors have unit length.)

Remark 1.   We mention two alternative descriptions of unit ℓ_2^2-representations that however will not play an explicit role in this article. (i) Geometrically speaking, the above triangle inequality says that every v_i and v_k subtend a nonobtuse angle at v_j. (ii) Every positive semidefinite $n \times n$ matrix has a *Cholesky factorization*, namely, a set of n vectors $v_1, v_2, \ldots, v_n$ such that $M_{ij} = \langle v_i, v_j \rangle$. Thus, a unit-$\ell_2^2$-representation for an n node graph can be alternatively viewed as a positive semidefinite $n \times n$ matrix M whose diagonal entries are 1 and $\forall i, j, k, M_{ij} + M_{jk} - M_{ik} \leq 1$.

We note that an ℓ_2^2-representation of a graph defines a so-called ℓ_2^2 metric on the vertices, where $d(i, j) = |v_i - v_j|^2$. We'll now see how the geometry of ℓ_2^2 relates to the properties of cuts in a graph.

Every cut $(S, \overline{S})$ gives rise to a natural unit-ℓ_2^2-representation, namely, one that assigns some unit vector v_0 to every vertex in S and $-v_0$ to every vertex in $\overline{S}$.

Thus, the following SDP is a relaxation for $\alpha_c(G)$ (scaled by cn).

$$\min \quad \frac{1}{4} \sum_{\{i,j\} \in E} |v_i - v_j|^2 \tag{4}$$

$$\forall i \quad |v_i|^2 = 1 \tag{5}$$

$$\forall i, j, k \quad |v_i - v_j|^2 + |v_j - v_k|^2 \geq |v_i - v_k|^2 \tag{6}$$

$$\sum_{i<j} |v_i - v_j|^2 \geq 4c(1 - c)n^2 \tag{7}$$

This SDP motivates the following definition.

Definition 2. An ℓ_2^2-representation is *c-spread* if equation (7) holds.

Similarly, the following is a relaxation for SPARSEST CUT (up to scaling by *n*; see Section 6).

$$\min \quad \sum_{\{i,j\}\in E} |v_i - v_j|^2 \tag{8}$$

$$\forall i,j,k \quad |v_i - v_j|^2 + |v_j - v_k|^2 \geq |v_i - v_k|^2 \tag{9}$$

$$\sum_{i<j} |v_i - v_j|^2 = 1 \tag{10}$$

As we mentioned before, the SDPs subsume both the eigenvalue approach and the Leighton–Rao approach [Goemans 1998]. We show that the optimum value of the SPARSEST CUT SDP is $\Omega(\alpha(G)n/\sqrt{\log n})$, which shows that the integrality gap is $O(\sqrt{\log n})$.

2.1. MAIN THEOREM ABOUT ℓ_2^2-REPRESENTATIONS. In general, ℓ_2^2-representations are not well-understood.[3] This is not surprising since in $\Re^d$ the representation can have at most 2^d distinct vectors [Danzer and Branko 1962], so our three-dimensional intuition is of limited use for graphs with more than 2^3 vertices. The technical core of our article is a new theorem about unit ℓ_2^2-representations.

Definition 3 (Δ-*separated*). If $v_1, v_2, \ldots, v_n \in \Re^d$, and $\Delta \geq 0$, two disjoint sets of vectors S, T are Δ-*separated* if for every $v_i \in S$, $v_j \in T$, $|v_i - v_j|^2 \geq \Delta$.

THEOREM 1 (MAIN). *For every* $c > 0$, *there are* $c', b > 0$ *such that every c-spread unit-ℓ_2^2-representation with n points contains Δ-separated subsets S, T of size* $c'n$, *where* $\Delta = b/\sqrt{\log n}$. *Furthermore, there is a randomized polynomial-time algorithm for finding these subsets* S, T.

Remark 2. The value of Δ in this theorem is best possible up to a constant factor, as demonstrated by the natural embedding (scaled to give unit vectors) of the Boolean hypercube $\{-1, 1\}^d$. These vectors form a unit ℓ_2^2-representation, and the isoperimetric inequality for hypercubes shows that every two subsets of $\Omega(n)$ vectors contain a pair of vectors — one from each subset — whose squared distance is $O(1/\sqrt{\log n})$ (here $n = 2^d$). Theorem 1 may thus be viewed as saying that every $\Omega(1)$-spread unit ℓ_2^2-representation is "hypercube-like" with respect to the property mentioned in the theorem. Some may view this as evidence for the conjecture that ℓ_2^2-representations are indeed like hypercubes (in particular, embed in ℓ_1 with low distortion). The recent results in Chawla et al. [2005] and Arora et al. [2008], which were inspired by our article, provide partial evidence for this belief.

2.1.1. *Corollary to Main Theorem, $\sqrt{\log n}$-approximation.* Let $W = \sum_{\{i,j\}\in E} \frac{1}{4}|v_i - v_j|^2$ be the optimum value for the SDP defined by Eqs. (4)–(7). Since the vectors v_i's obtained from solving the SDP satisfy the hypothesis of Theorem 1,

---

[3] As mentioned in the introduction, it has been conjectured that they are closely related to the better-understood ℓ_1 metrics.
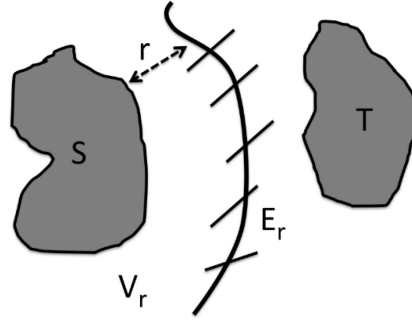
FIG. 1. V_r is the set of nodes whose distance on the weighted graph to S is at most r. Set of edges leaving this set is E_r.

as an immediate corollary to the theorem we show how to produce a c'-balanced cut of size $O(W\sqrt{\log n})$. This uses the region-growing argument of Leighton–Rao.

COROLLARY 2. *There is a randomized polynomial-time algorithm that finds with high probability a cut that is c'-balanced, and has size $O(W\sqrt{\log n})$.*

PROOF. We use the algorithm of Theorem 1 to produce Δ-separated subsets S, T for $\Delta = b/\sqrt{\log n}$. Let V_0 denote the vertices whose vectors are in S. Associate with each edge $e = \{i, j\}$ a length $w_e = \frac{1}{4}|v_i - v_j|^2$. (Thus, $W = \sum_{e \in E} w_e$.) S and T are at least Δ apart with respect to this distance. Let V_s denote all vertices within distance s of S. Now we produce a cut as follows: pick a random number r between 0 and Δ, and output the cut $(V_r, V - V_r)$. Since $S \subseteq V_r$ and $T \subseteq V \setminus V_r$, this is a c'-balanced cut. (See Figure 1)

To bound the size of the cut, we denote by E_r the set of edges leaving V_r. Each edge $e = \{i, j\}$ only contributes to E_r for r in the open interval (r_1, r_2), where $r_1 = d(i, V_0)$ and $r_2 = d(j, V_0)$. Triangle inequality implies that $|r_2 - r_1| \le w_e$. Hence

$$W = \sum_e w_e \ge \int_{s=0}^{\Delta} |E_s|.$$

Thus, the expected value of $|E_r|$ over the interval $[0, \Delta]$ is at most W/Δ. The algorithm thus produces a cut of size at most $2W/\Delta = O(W\sqrt{\log n})$ with probability at least $1/2$. □

3. $\Delta = \Omega(\text{Log}^{-2/3} n)$-*Separated Sets*

We now describe an algorithm that given a c-spread ℓ_2^2 representation finds Δ-separated sets of size $\Omega(n)$ for $\Delta = \Theta(1/\log^{2/3} n)$. Our correctness proof assumes a key lemma (Lemma 7) whose proof appears in Section 4. The algorithm will be improved in Section 5 to allow $\Delta = \Theta(1/\sqrt{\log n})$.

The algorithm, SET-FIND, is given a c-spread ℓ_2^2-representation. Constants $c', \sigma > 0$ are chosen appropriately depending on c.
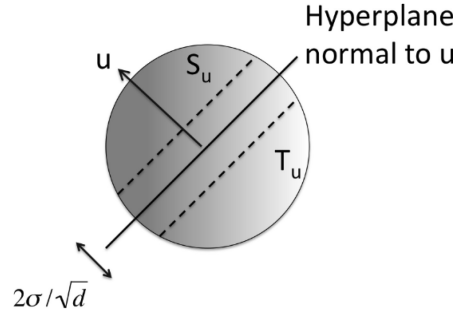
FIG. 2. The sets S_u and T_u found during SET-FIND. The figure assumes $m = 0$.

SET-FIND:

Input: A c-spread unit-ℓ_2^2 representation $v_1, v_2, \ldots, v_n \in \mathfrak{R}^d$.

Parameters: Desired separation Δ, desired balance c', and projection gap, σ.

Step 1: Project the points on a uniformly random line u passing through the origin, and compute the largest value m where half the points v, have $\langle v, u \rangle \geq m$. (See Figure 2) Then, we specify that

$$S_u = \left\{ v_i : \langle v_i, u \rangle \geq m + \frac{\sigma}{\sqrt{d}} \right\},$$

$$T_u = \{ v_i : \langle v_i, u \rangle \leq m \}.$$

If $|S_u| < 2c'n$, HALT.[a]

Step 2: Pick any $v_i \in S_u, v_j \in T_u$ such that $\left| v_i - v_j \right|^2 \leq \Delta$, and delete v_i from S_u and v_j from T_u. Repeat until no such v_i, v_j can be found and output the remaining sets S, T.

---

[a] We note that we could have simply chosen m to be 0 rather than to be the median, but this version of the procedure also applies for finding sparsest cuts.

Remark 3. The procedure SET-FIND can be seen as a rounding procedure of sorts. It starts with a "fat" random hyperplane cut (cf. Goemans–Williamson [1995]) to identify the sets S_u, T_u of vertices that project far apart. It then prunes these sets to find sets S, T.

Notice that if SET-FIND does not HALT prematurely, it returns a Δ-separated pair of sets. It is easy to show that the likelihood of premature halting is small, and we defer it to the next section. The main challenge is to show that that the Δ-separated sets are large, that is, that no more than $c'n$ pairs of points are deleted from S_u and T_u. This occupies the bulk of the article and we start by giving a very high level picture.

3.1. BROAD OUTLINE OF PROOF. First, observe that the Theorem is almost trivial if we desire $O(\log n)$-separated sets S and T (in other words, $\Delta = \Omega(1/\log n)$). The reason is that the expected projection of a vector of length ℓ is $\ell/\sqrt{d}$ and the chance that it is k times larger is $\exp(-k^2)$ (see Lemma 5). Since a pair is deleted only if the projection of $v_i - v_j$ on u exceeds $\Omega(1/\sqrt{d})$, the chance that a pair satisfying $|v_i - v_j| = O(1/\sqrt{\log n})$ or $|v_i - v_j|^2 = O(1/\log n)$ is deleted

is $\exp(-O(\log n))$, which is polynomially small. So, with high probability *no pair* is deleted. Unfortunately, this reasoning breaks for the case of interest, namely $\Delta = \Omega(\log^{-2/3} n)$, when many pairs may get deleted.

So assume the algorithm fails with high probability when $\Delta = \Omega(\log^{-2/3} n)$. We show that for most directions u there is a sequence of $k = \log^{1/3} n$ points $v_1, v_2, \ldots, v_k$ such that every successive pair is close, that is, $|v_i - v_{i+1}|^2 \leq \Delta$, and their difference has a large projection on u, that is, $\langle v_i - v_{i+1}, u \rangle \geq 2\sigma/\sqrt{d}$. The main point is that these projections all have the same *sign*, and thus, adding them we obtain $\langle v_1 - v_k, u \rangle \geq 2\sigma k/\sqrt{d}$. Thus, the projection scales linearly in k whereas the Euclidean distance between the first and last point in the sequence scales as $\sqrt{k}$, since the points come from an ℓ_2^2 metric. For the chosen values of k, Δ, this means that Euclidean distance between the first and last point is $O(1)$ whereas the projection is $\Omega(\sqrt{\log n}/\sqrt{d})$ — large enough that the probability that such a large projection exists for *any* of the $\binom{n}{2}$ pairs is $o(1)$. This is a contradiction, allowing us to conclude that the algorithm did not fail with high probability (in other words, did manage to output large Δ-separated sets).

The idea in finding this sequence of points is to find them among deleted pairs (corresponding to different directions u). This uses the observation that the deleted pairs for a direction u form a matching, and that the path for a direction u can in fact use deleted pairs from a "nearby" direction u'; something that is made formal using measure concentration. The argument uses induction, and actually shows the existence of many sequences of the type described above, not just one.

3.2. COVERS AND MATCHING COVERS.   Let us start by formalizing the condition that must hold if the algorithm fails with high probability, that is, for most choices of directions u, $\Omega(n)$ pairs must be deleted. It is natural to think of the deleted pairs as forming a matching.

Definition 4.   A (σ, δ, c')-*matching cover* of a set of points in $\Re^d$ is a set $\mathcal{M}$ of matchings such that for at least a fraction δ of directions u, there exists a matching $M_u \in \mathcal{M}$ of at least $c'n$ pairs of points, such that each pair $(v_i, v_j) \in M_u$, satisfies

$$\langle v_i - v_j, u \rangle \geq 2\sigma/\sqrt{d}.$$

The associated *matching graph* M is defined as the multigraph consisting of the union of all matchings M_u.

In the next section, we will show that Step (1) succeeds with probability δ. This implies that if with probability at least $\delta/2$ the algorithm did *not* produce sufficiently large Δ-separated pair of sets, then with probability at least $\delta/2$ many pairs were deleted. Thus, the following lemma is straightforward.

LEMMA 3.   *If* SET-FIND *fails with probability greater than* $1 - \delta/2$, *on a unit c-spread ℓ_2^2-representation, then the set of points has a $(\sigma, \delta/2, c')$-matching cover.*

The definition of (σ, δ, c')-matching cover $\mathcal{M}$ of a set of points suggests that for many directions there are many disjoint pairs of points that have long projections. We will work with a related notion.

Definition 5 ((σ, δ)-*uniform-matching-cover*).   A set of matchings $\mathcal{M}$ (σ, δ)-*uniform-matching-covers* a set of points $V \subseteq \Re^d$ if for every unit vector $u \in \Re^d$, there is a matching M_u of V such that every $(v_i, v_j) \in M_u$ satisfies $\left|\langle u, v_i - v_j \rangle\right| \geq$

$\frac{2\sigma}{\sqrt{d}}$, and for every i, $\mu(u : v_i$ matched in $M_u) \geq \delta$. We refer to the set of matchings $\mathcal{M}$ to be the *matching cover of V*.

Remark 4.   The main difference from Definition 4 is that every point participates in M_u with constant probability for a random direction u, whereas in Definition 4 this probability could be even 0.

LEMMA 4.   *If a set of n vectors is (σ, γ, β)-matching covered by $\mathcal{M}$, then they contain a subset X of $\Omega(n\delta)$ vectors that are (σ, δ)-uniformly matching covered by $\mathcal{M}$, where $\delta = \Omega(\gamma\beta)$.*

PROOF.   Consider the multigraph consisting of the union of all matchings M_u's as described in Definition 4. The average node is in M_u for at least $\gamma\beta$ measure of directions. Remove all nodes that are matched on fewer than $\gamma\beta/2$ measure of directions (and remove the corresponding matched edges from the M_u's). Repeat. The aggregate measure of directions removed is at most $\gamma\beta n/2$. Thus, at least $\gamma\beta n/2$ aggregate measure on directions remains. This implies that there are at least $\gamma\beta n/4$ nodes left, each matched in at least $\gamma\beta/4$ measure of directions. This is the desired subset X.   □

To carry out our induction we need the following weaker notion of covering a set:

Definition 6 $((\epsilon, \delta)$-*cover*).   A set $\{w_1, w_2, \ldots, \}$ of vectors in $\Re^d$ is an (ϵ, δ)-*cover* if every $|w_j| \leq 1$ and for at least δ fraction of unit vectors $u \in \Re^d$, there exists a j such that $\langle u, w_j \rangle \geq \epsilon$. We refer to δ as the *covering probability* and ϵ as the *projection length* of the cover.
    A point v is (ϵ, δ)-covered by a set of points X if the set of vectors $\{x - v | x \in X\}$ is an (ϵ, δ)-cover.

Remark 5.   It is important to note that matching covers are substantially stronger notion than covers. For example, the d-dimensional hypercube $\frac{1}{\sqrt{d}}\{-1, 1\}^d$ is easily checked to be an $(\Omega(1), \Omega(1))$-cover (equivalently, the origin is $(\Omega(1), \Omega(1))$ covered), even though it is only $(\Theta(1), \Theta(1/\sqrt{d}))$ uniformly matching covered. This example shows how a single outlier in a given direction can cover all other points, but this is not sufficient for a matching cover.

3.3. GAUSSIAN BEHAVIOR OF PROJECTIONS.   Before we get to the main lemma and theorem, we introduce a basic fact about projections and use it to show that Step (1) of SET-FIND succeeds with good probability.

LEMMA 5 (GAUSSIAN BEHAVIOR OF PROJECTIONS) *If v is a vector of length ℓ in $\Re^d$ and u is a randomly chosen unit vector, then*

(1) *for $x \leq 1$, $\Pr[|\langle v, u \rangle| \leq \frac{x\ell}{\sqrt{d}}] \leq 3x$.*

(2) *for $x \leq \sqrt{d}/4$, $\Pr[|\langle v, u \rangle| \geq \frac{x\ell}{\sqrt{d}}] \leq e^{-x^2/4}$.*

Formally, to show that Step (1) of SET-FIND succeeds with good probability, we prove that the point sets satisfy conditions in the following definition.

Definition 7.   A set of n points $\{v_1, \ldots, v_n\} \in \Re^d$ is (σ, γ, c')-projection separated if for at least γ fraction of the directions u, $|S_u| \geq 2c'n$, where

$$S_u = \{i : \langle v_i, u \rangle \geq m + \sigma/\sqrt{d}\},$$

with m being the median value of $\{\langle v_i, u \rangle\}$.

Using part (i) of Lemma 5, we show that the points are sufficiently projection separated. Here, it is convenient for the points to be on the surface of the unit sphere. Later, for approximating SPARSEST CUT we prove a similar statement without this condition.

LEMMA 6.   *For every positive $c < 1/3$, there are $c', \sigma, \delta > 0$, such that every c-spread unit ℓ_2^2-representation is (σ, δ, c')-projection separated.*

PROOF.   The c-spread condition states that $\sum_{k,j} |v_k - v_j|^2 \geq 4c(1-c)n^2$. Since $|v_k - v_j| \leq 2$, we can conclude that $\sum_{k,j} |v_k - v_j| \geq 2c(1-c)n^2$. Applying Markov's inequality, we can conclude that $|v_i - v_j| \geq c(1-c)$ for at least $c(1-c)n^2$ pairs.

For these pairs i, j, condition (1) of Lemma 5 implies that for a random u, that $|\langle v_i, u \rangle - \langle v_j, u \rangle|$ is at least $c(1-c)/(9\sqrt{d})$ with probability at least $1/3$.

This implies that for $\sigma = c(1-c)/18$, the expected number of pairs where $|\langle v_i, u \rangle - \langle v_j, u \rangle| \geq 2\sigma/\sqrt{d}$ is at least $(2/3)c(1-c)n^2$. Each such pair must have at least one endpoint whose projection is at least $\sigma/\sqrt{d}$ away from m; and each point can participate in at most n such pairs. Therefore, the expected number of points projected at least $\sigma/\sqrt{d}$ away from m is at least $(2/3)c(1-c)n$.

Applying Markov's bound (and observing that the number of points is at most n) it follows that with probability at least $(1/3)c(1-c)$ at least $(1/3)c(1-c)n$ points are projected at least $\sigma/\sqrt{d}$ away from m. By symmetry, half the time a majority of these points are in S_u. Thus, the lemma follows with $\delta = c' = (1/6)c(1-c)$. $c' = (1/6)c(1-c)$.   □

3.4. MAIN LEMMA AND THEOREM.   We are now ready to formally state a lemma about the existence, for most directions, of many pairs of points whose difference $v_i - v_j$ has a large projection on this direction. (Sometimes, we will say more briefly that the "pair has large projection.") For each k, we inductively ensure that there is a large set of points which participate as one endpoint of such a (large projection) pair for at least half the directions. This lemma (and its variants) is a central technical contribution of this article.

Definition 8.   Given a set of n points that is (σ, δ, c')-matching covered by $\mathcal{M}$ with associated matching graph M, we define v to be in the *k-core, S_k,* if v is $(k\frac{\sigma}{2\sqrt{d}}, 1/2)$-covered by points that are within k hops of v in the matching graph M.

LEMMA 7.   *For every set of n points $\{v_1, \ldots, v_n\}$ that is (σ, δ, c')-matching covered by $\mathcal{M}$ with associated matching graph M, there are positive constants $a = a(\delta, c')$ and $b = b(\delta, c')$ such that for every $k \geq 1$:*

(1) *Either $|S_k| \geq a^k n$.*

(2) *Or there is a pair (v_i, v_j) with distance at most k in the matching graph M, such that*

$$\left|v_i - v_j\right| \geq b\sigma/\sqrt{k}.$$

We use this lemma below to show that SET-FIND finds $\Omega(1/\log^{1/3} n)$-separated sets. Later, we develop a modified algorithm for finding $\Omega(1/\sqrt{\log n})$-separated sets.

Subsequently, the aforementioned paper by J. R. Lee [2005] strengthened Case (1) of the lemma to show the existence of S_k of size at least $|M|/2$, and thus that Case (2) meets the stronger condition that $\left|v_i - v_j\right| \geq g\sigma$ for some constant g. This implies that SET-FIND without any modification finds large $\Omega(1/\sqrt{\log n})$-separated sets.

We can now prove the following theorem.

THEOREM 8. *Set-find finds an Δ-separated set for a c-spread unit ℓ_2^2-representation $\{v_1, \ldots, v_n\}$ with constant probability for some $\Delta = \Omega(1/\log^{2/3} n)$.*

PROOF. Recall, that, by Lemma 3, if SET-FIND fails, then the set of points is (σ, δ, c')-matching covered. Moreover, each pair (v_i, v_j) in any matching M_u for a direction u satisfies $\left|v_i - v_j\right| \leq \sqrt{\Delta}$ and $\langle v_i - v_j, u \rangle \geq 2\sigma/\sqrt{d}$.

We will now apply Lemma 7 to show that such a matching covered set does not exist for $\Delta = \Theta(1/\log^{2/3} n)$, which means that SET-FIND cannot fail for this Δ. Concretely, we show that neither case of the Lemma holds for $k = b\sigma/\sqrt{\Delta}$ where b is the constant defined in the Lemma.

Let us start by dispensing with Case (2). Since the metric is ℓ_2^2, a pair of points (v_i, v_j) within distance k in the matching graph satisfies $|v_i - v_j|^2 \leq k\Delta$, and thus $|v_i - v_j| \leq \sqrt{k\Delta}$. For our choice of k, this is less than $b\sigma/\sqrt{k}$, so Case (2) does not happen.

Case (1) requires that for at least $\frac{1}{2}$ of the directions u there is a pair of vectors (v_i, v_j) such that $|v_i - v_j| \leq \sqrt{k\Delta}$ and where $\langle v_i - v_j, u \rangle \geq k\sigma/2\sqrt{d}$.

For any fixed i, j, the probability that a randomly chosen direction satisfies this condition is at most $\exp(-k\sigma^2/16\Delta)$, which can be made smaller than $1/2n^2$ by choosing some $\Delta = \Omega(1/\log^{2/3} n)$ since $k = \Theta(\log^{1/3} n)$. Therefore, by the union bound, the fraction of directions for which any pair i, j satisfies this condition is less than $n^2/2n^2 \leq 1/2$. Thus, Case (1) also cannot happen for this value of Δ.

This proves the theorem. $\square$

4. *Proof of Lemma 7*

4.1. MEASURE CONCENTRATION. We now introduce an important geometric property of covers. If a point is covered in a non-negligible fraction of directions, then it is also covered (with slightly smaller projection length) in almost all directions. This property follows from measure concentration, which can be stated as follows.

Let S^{d-1} denote the surface of the unit ball in $\Re^d$ and let $\mu(\cdot)$ denote the standard measure on it. For any set of points A, we denote by A_γ the γ-*neighborhood of A,* namely, the set of all points that have distance at most γ to some point in A.

LEMMA 9 (CONCENTRATION OF MEASURE) *If $A \subseteq S^{d-1}$ is measurable and $\gamma > \frac{2\sqrt{\log(1/\mu(A))}+t}{\sqrt{d}}$, where $t > 0$, then $\mu(A_\gamma) \geq 1 - \exp(-t^2/2)$.*

PROOF. Levy's isoperimetric inequality (see Ball [1997]) states that $\mu(A_\gamma)/\mu(A)$ is minimized for spherical caps.[4] The lemma now follows by a simple calculation using the standard formula for (d-1)-dimensional volume of spherical caps, which says that the cap of points whose distance is at least $s/\sqrt{d}$ from an equatorial plane is $\exp(-s^2/2)$. □

The following lemma applies measure concentration to boost covering probability.

LEMMA 10 (BOOSTING LEMMA) *Let $\{v_1, v_2, \ldots\}$ be a finite set of vectors that is an (ϵ, δ)-cover, and $|v_i| \leq \ell$. Then, for any $\gamma > \frac{\sqrt{2\log(2/\delta)}+t}{\sqrt{d}}$, the vectors are also an $(\epsilon - \ell\gamma, \delta')$-cover, where $\delta' = 1 - \exp(-t^2/2)$.*

PROOF. Let A denote the set of directions u for which there is an i such that $\langle u, v_i \rangle \geq \epsilon$. Since $|v_i - u|^2 = 1 + |v_i|^2 - 2\langle u, v_i \rangle$ we also have:

$$A = S^{d-1} \cap \bigcup_i \text{Ball}\left(v_i, \sqrt{1 + |v_i|^2 - 2\epsilon}\right),$$

which also shows that A is measurable. Also, $\mu(A) \geq \delta$ by hypothesis. Thus, by Lemma 9, $\mu(A_\gamma) \geq 1 - \exp(-t^2/2)$.

We argue that for each direction u in A_γ, there is a vector v_i in the (ϵ, δ) cover with $\langle v_i, u \rangle \geq \epsilon - 2\ell\gamma$ as follows. Let $u \in A$, $u' \in A_\gamma \cap S^{d-1}$ be such that $|u - u'| \leq \gamma$.

We observe that $\langle u', v_i \rangle = \langle u, v_i \rangle + \langle u' - u, v_i \rangle \geq \epsilon - \gamma\ell$, since $|v_i| \leq \ell$.

Combined with the lower bound on $\mu(A_\gamma)$, we conclude that the set of directions u' such that there is an i such that $\langle u', v_i \rangle \geq \epsilon - 2\ell\gamma$ has measure at least $1 - \exp(-t^2/2)$. □

4.2. COVER COMPOSITION. In this section, we give the construction that lies at the heart of the inductive step in the proof of Lemma 7.

Let $X \subseteq \Re^d$ be a point set that is uniform (ϵ, δ)-matching-covered, where M_u denotes as usual the matching associated with direction u. Suppose $Z \subseteq X$ consists of points in X that are $(\epsilon_1, 1 - \delta/2)$-covered, and for $v_i \in Z$ let Z_i denote points that cover v_i.

The following lemma shows how to combine the cover and matching cover to produce a cover for a set Z' with projection length the sum of the projection lengths of the original covers.

LEMMA 11 (COVER COMPOSITION) *If $|Z| \geq \tau |X|$, then there is a set Z' consisting of at least $\delta\tau/4$ fraction of points in X that are $(\epsilon_1 + \epsilon, \delta\tau/4)$-covered (see Figure 3). Furthermore, the cover for each such point v_k can even be found with*

---

[4] Levy's isoperimetric inequality is not trivial; see Schechtman [2003] for a sketch. However, results qualitatively the same— but with worse constants —as Lemma 9 can be derived from the more elementary Brunn-Minkowski inequality; this "approximate isoperimetric inequality" of Ball, de Arias and Villa also appears in Schechtman [2003].
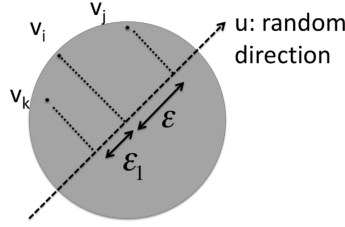
FIG. 3.   Direction u is chosen randomly. The vector $v_j - v_k$ has projection $\epsilon_1 + \epsilon$ on u and it gets "assigned" to k.

only vectors $v_j - v_k$ that can be expressed as

$$v_j - v_k = v_j - v_i + v_i - v_k,$$

where (i) $v_j - v_i \in Z_i$ (ii) $\{v_i, v_j\}$ is an edge in M_u for some direction u.

Remark 6.   This is called "cover composition" because each vector $v_j - v_k$ in the new cover can be written as a sum of two vectors, one of which is in the matching cover and the other in the cover of v_i.

PROOF.   Let $v_i \in Z$. For $1 - \delta/2$ fraction of directions u, there is a point $v_j \in Z_i$ such that $\langle v_j - v_i, u \rangle \geq \epsilon_1$. Also for δ fraction of directions u, there is a point in $v_k \in X$ such that $\langle v_k - v_i, u \rangle \leq -\epsilon$ and v_k is matched to v_i in the matching M_u. Thus for a $\delta/2$ fraction of directions u, both events happen and thus the pair (v_j, v_k) satisfies $\langle v_j - v_k, u \rangle \geq \epsilon_1 + \epsilon$. Since $v_j \in Z_i$ and $v_k \in X$, we "assign" this vector $v_j - v_k$ to point v_k for direction u, as a step towards building a cover centered at v_k. Now we argue that for many v_k's, the vectors assigned to it in this way form an $(\epsilon_1 + \epsilon, \delta |Z|/2 |X|)$-cover.

For each point $v_i \in Z$, for $\delta/2$ fraction of the directions u the process above assigns a vector to a point in X for direction u according to the matching M_u. Thus, on average for each direction u, at least $\delta|Z|/2$ vectors get assigned by the process. Thus, for a random point in X, the expected measure of directions for which the point is assigned a vector is at least $\delta|Z|/2|X|$.

Furthermore, at most, one vector is assigned to any point for a given direction u (since the assignment is governed by the matching M_u). Therefore, at least $\delta |Z|/4 |X|$ fraction of the points in X must be assigned a vector for $\delta |Z|/4 |X| = \delta\tau/4$ fraction of the directions.

We will define all such points of X to be the set Z' and note that the size is at least $\delta |Z|/4$ as required.   □

For the inductive proof of Lemma 7, it is useful to boost the covering probability of the cover obtained in Lemma 11 back up to $1 - \delta/2$. This is achieved by invoking the boosting lemma as follows.

COROLLARY 12.   *If the vectors in the covers for Z' have length at most*

$$\frac{\epsilon\sqrt{d}}{4\sqrt{\log \frac{8}{\tau\delta}} + 2\sqrt{\log \frac{2}{\delta}}},$$

then the $(\epsilon_1 + \epsilon, \tau\delta/4)$-covers for Z' are also $(\epsilon_1 + \epsilon/2, 1 - \delta/2)$-covers.

PROOF.   This follows from Lemma 10, since the loss in projection in this corollary is $\epsilon/2$, which should be less than $\gamma\ell$ where ℓ is the length of the vectors in the cover and γ is defined as in Lemma 10. Solving for ℓ yields the upper bound in the corollary.   □

Though the upper bound on the length in the hypothesis appears complicated, we only use the lemma with $\epsilon = \Theta(1/\sqrt{d})$ and constant δ. Thus, the upper bound on the length is $\Theta(1/\sqrt{\log(1/\tau)})$. Furthermore, $1/\tau$ will often be $\Theta(1)$.

4.3. PROOF OF LEMMA 7.   The proof of the lemma is carried out using Lemma 10 and Corollary 12 in a simple induction.

Before proving the lemma, let us restate a slight strengthening to make our induction go through.

Definition 9.   Given a set of n points that is (σ, δ, c')-matching covered by $\mathcal{M}$ with associated matching graph M, we define v to be in the (k, ρ)-*core*, $S_{k,\rho}$, if v is $(k\frac{\sigma}{2\sqrt{d}}, 1-\rho)$-covered by points that are within k hops of v in the matching graph M.

CLAIM 13.   *For constant δ, and every $k \geq 1$,*

(1) $|S_{k,\delta/2}| \geq (\frac{\delta}{4})^k|X|$

(2) *or there is a pair (v_i, v_j) such that $\left| v_i - v_j \right| = \Omega(\sigma/\sqrt{k})$ and v_i and v_j are within k hops in the matching graph M.*

PROOF.   Since the set of points is (σ, δ, c')-matching covered, Lemma 4 implies that there exists a (σ, δ)-uniform matching cover $\mathcal{M}$ of a subset X of the point set. For $k = 1$, the claim follows by setting $S_{1,\delta/2} = X$.

Now we proceed by induction on k.

Clearly if case (2) holds for k, then it also holds for $k + 1$. So assume that case (1) holds for k, that is, $S_{k,\delta/2}$ satisfies the conditions of case (1). Composing the covers of $S_{k,\delta/2}$ with the matching cover $\mathcal{M}$ using Lemma 11 yields a cover for a set Z' of size at least $\frac{\delta}{4}|S_k|$ with covering probability $\Theta(|S_k|/|X|)$, but with projection length that is larger by $\epsilon = \Omega(1/\sqrt{d})$. To finish the induction step the covering probability must be boosted to $1 - \delta/2$ that, by Lemma 10, decreases the projection length by

$$\ell\sqrt{\log\Theta(|X|/|S_k|)}/\sqrt{d} = O(\ell\sqrt{k}/\sqrt{d}), \tag{11}$$

where ℓ upper-bounds the length of the vectors in the cover. If this decrease is less than $\epsilon/2$, then the points in Z' are $(k\epsilon/2 + \epsilon/2, 1 - \delta/2)$-covered and $S_{k+1,\delta/2}$ is large as required in case 1. Otherwise we have $\epsilon/2 = \Theta(\sigma/\sqrt{d}) = O(\ell\sqrt{k}/\sqrt{d})$, which simplifies to $\ell \geq g\sigma/\sqrt{k}$, for some constant g and thus case 2 holds for $k + 1$.   □

Remark 7.   An essential aspect of this proof is that the k-length path (from M^k) pieces together pairs from many different matchings M_u. Consider, for example, given a direction u and a matching M_u with projection ϵ on u, how we produce any pair of points with larger projection on u. Clearly, edges from the matching M_u do not help with this. Instead, given any pair $(x, y) \in M_u$, we extend it with a pair

$(y, z) \in M_{u'}$ for some different direction u' where (y, z) also has a large projection on u. The existence of such a pair (y, z) follows from the boosting lemma.

5. *Achieving* $\Delta = \Omega(1/\sqrt{\text{Log } n})$

Theorem 1 requires $\Delta = \Omega(1/\sqrt{\log n})$ whereas we proved above that SET-FIND finds large Δ-separated sets for $\Delta = \Omega(\log^{-2/3} n)$. Now we would like to run SET-FIND for $\Delta = \Omega(1/\sqrt{\log n})$. The main problem in proving that SET-FIND succeeds for larger Δ is that our version of Lemma 7 is too weak. Before seeing how to remedy this it is useful to review how the proof of Theorem 1 constrains the different parameters and results in a $\Theta(1/\log^{2/3} n)$ upper boundon Δ.

(1) In the proof of Lemma 7, we can continue to inductively assert case (1) (producing covered set S_k) of Claim 13 as long as the right-hand side of Eq. (11) is less than $\sigma/2\sqrt{d}$, that is, as long as $\ell^2 \ll 1/k$, where ℓ bounds the lengths of the vectors.
(2) By the triangle inequality, $\ell = O(\sqrt{k\Delta})$.
(3) Deriving the final contradiction in the proof of Theorem 8 requires that $\exp(-k/\Delta) \ll 1/n^2$.

The limit on the vector lengths in item (1), combined with item (2), requires that $k \leq 1/\Delta^{1/2}$, and thus item (3) constrains Δ to be $O(1/\log^{2/3} n)$.

The improvement derives from addressing the limit on vector lengths in item 1, which arises from the need to boost the covering probability for S_k (whose size decreases exponentially in k) from $\Theta(|S_k|/|X|)$ to $1 - \delta/2$ as detailed in Eq. (11). Thus, we would like to prevent the size of S_k from decreasing below $\Omega(|X|)$. This will allow the induction to continue until the vectors have length $\Omega(1)$. To do so, we will use the fact that any point close to a point which is (ϵ, δ)-covered is also (ϵ', δ')-covered where $\epsilon' \approx \epsilon$ and $\delta' \approx \delta$. This is quantified in Lemma 14.

LEMMA 14 (COVERING CLOSE POINTS). *Suppose* $v_1, v_2, \ldots \in \Re^d$ *form an* (ϵ, δ)-*cover for* v_0. *Then, they also form an* $(\epsilon - \frac{ts}{\sqrt{d}}, \delta - e^{-t^2/4})$-*cover for every* v_0' *such that* $\left| v_0 - v_0' \right| \leq s$.

PROOF. If u is a random unit vector, $\Pr_u[\langle u, v_0 - v_0' \rangle \geq \frac{ts}{\sqrt{d}}] \leq e^{-t^2/4}$. □

Remark 8. Lemma 14 shows that (ϵ, δ) covers of a point are remarkably robust. In our context, where we can afford to lose $\Theta(1/\sqrt{d})$ in projection length, even a $\Theta(1)$-neighborhood of that point is well covered.

Definition 10 (ζ-*proximate graph*). A graph G on a set of points $v_1, \ldots, v_n \in \Re^d$ is called ζ-*proximate* if for each edge (v_i, v_j) in G, satisfies $\left| v_i - v_j \right| \leq \zeta$. A set of vertices S is *nonmagnifying* if $|S \cup \Gamma(S)| < |V|/2$, where $\Gamma(S)$ is the set of neighbors of S in G.

Notice that, for any set S, if the ζ-proximate graph is chosen to have edges between all pairs with distance at most ζ that if $T = V - S - \Gamma(S)$ then S, T is a ζ^2-separated pair of sets. We will use $\zeta^2 = \Delta$, thus, if we ever find a nonmagnifying set S of size $\Omega(n)$ then we are done.

We prove a stronger version of Lemma 7 where case (2) only occurs when vectors have $\Omega(1)$ length by ensuring in case (1) that the $(k, \delta/2)$-core, $S_{k,\delta/2}$, has size at least $|X|/2$.

The induction step in Lemma 11 now yields a set, which we denote by S'_{k+1}, of cardinality $\delta|X|$, which is covered by X with constant covering probability and with projection length increased by $\Omega(1/\sqrt{d})$. If S'_{k+1} is a non-magnifying set in the ζ-proximate graph, we can halt (we have found a ζ-separated pair of sets for the ζ-proximate graph above). Otherwise, S'_{k+1} can be augmented with $\Gamma(S'_{k+1})$ to obtain a set T of size $|X|/2$, which, by Lemma 14, is covered with projection length that is smaller by $O(\zeta/\sqrt{d})$ and is contained in $S_{k+1,\delta/2}$.

Then, we boost the covering probability at a cost of reducing the projection length by $O(\ell/\sqrt{d})$ where ℓ is the length of the covering vectors. Thus when $\zeta \ll 1$, we either increase the projection length by $\Omega(1/\sqrt{d})$ or ℓ is too large or we produce a nonmagnifying set. This argument yields the following enhanced version of Lemma 7:

LEMMA 15. *For every set of points X, that is (δ, σ)-uniform matching covered by $\mathcal{M}$ with associated matching graph M, and a ζ-proximate graph on the points for $\zeta \leq \zeta_0(\delta, c', \sigma)$, at least one of the following is true for every $k \geq 1$, where $g = g(\delta, c')$ is a positive constant.*

(1) *The $(k, \delta/2)$-core, $S_{k,\delta/2}$, has cardinality at least $|X|/2$*
(2) *There is a pair (v_i, v_j) with distance at most k in the matching graph M, such that*

$$\left| v_i - v_j \right| \geq g.$$

(3) *The set, $S'_{k'}$, is nonmagnifying in the ζ-proximate graph for some $k' < k$.*

We can now prove Theorem 1. We use set-find with some parameter $\Delta = \Theta(1/\sqrt{\log n})$. If it fails to find a Δ-separated set, the point set must have been (σ, δ, c')-matching covered where δ, c', and σ are $\Theta(1)$. We first apply Lemma 4 to obtain a (σ, δ')-uniform matching covered set X of size $\Omega(n)$. We then apply Lemma 15 where the edges in the ζ-proximate graph consist of all pairs in X whose ℓ_2^2 distance is less than some $\zeta^2 = \Theta(1/\sqrt{\log n})$. If case (3) ever occurs, we can produce a pair of ζ-separated of size $\Omega(n)$. Otherwise, we can continue the induction until $k = \Omega(\sqrt{\log n})$ at which point we have a cover with projection length $\Theta(\sqrt{\log n}/\sqrt{d})$ consisting of vectors of $O(1)$ length. As in our earlier proof, this contradicts the assumption that set-find failed.

Thus, either SET-FIND gives a Δ-separated pair of large sets, or for some k, the ζ-neighborhood of S_k, $\Gamma(S_k)$, is small and thus, S_k, $V - \Gamma(S'_k)$ forms a ζ-separated pair of large sets. Clearly, we can identify such an S_k by a randomized algorithm that uses random sampling of directions to check whether a vertex is well covered.

This proves Theorem 1.

Remark 9. Lee's direct proof that SET-FIND [2005] produces $\Omega(1/\sqrt{\log n})$-separated sets relies on the following clever idea: rather than terminating the induction with a nonmagnifying set (in the ζ-proximate graph), he shows how to use the nonmagnifying condition to bound the loss in covering probability. The main observation is that after cover composition, the covering probability is actually $|S_k|/|\Gamma(S_k)|$ rather than our pessimistic bound of $|S_k|/|X|$. Thus, by insisting on

the invariant that $|\Gamma(S_k)| = O(|S_k|)$, he ensures that the covering probability falls by only a constant factor, thus incurring only a small boosting cost and allowing the induction to continue for $k = \Omega(1/\Delta)$ steps. Maintaining the invariant is easy since S_k can be replaced by $\Gamma(S_k)$ with small cost in projection length using Lemma 14. The point being that over the course of the induction this region growing (replacing S_k by $\Gamma(S_k)$) step only needs to be invoked $O(\log n)$ times.

6. $O(\sqrt{\text{Log}\,n})$ *Ratio for* SPARSEST CUT

Now we describe a rounding technique for the SDP in (8)—(10) that gives an $O(\sqrt{\log n})$-approximation to SPARSEST CUT. Note that our results on expander flows in Section 7 give an alternative $O(\sqrt{\log n})$-approximation algorithm.

First we see in what sense the SDP in (8)—(10) is a relaxation for SPARSEST CUT. For any cut $(S, \overline{S})$ consider a vector representation that places all nodes in S at one point of the sphere of squared radius $(2|S||\overline{S}|)^{-1}$ and all nodes in $|\overline{S}|$ at the diametrically opposite point. It is easy to verify that this solution is feasible and has value $|E(S, \overline{S})|/|S||\overline{S}|$.

Since $|\overline{S}| \in [n/2, n]$, we conclude that the *optimal* value of the SDP multiplied by $n/2$ is a lower bound for SPARSEST CUT.

The next theorem implies that the integrality gap is $O(\sqrt{\log n})$.

THEOREM 16. *There is a polynomial-time algorithm that, given a feasible SDP solution with value β, produces a cut $(S, \overline{S})$ satisfying $\left|E(S, \overline{S})\right| = O(\beta |S| n\sqrt{\log n})$.*

The proof divides into two cases, one of which is similar to that of Theorem 1. The other case is dealt with in the following Lemma.

LEMMA 17. *For every choice of constants c, τ where $c < 1, \tau < 1/8$, there is a polynomial-time algorithm for the following task. Given any feasible SDP solution with $\beta = \sum_{\{i,j\}\in E} \left|v_i - v_j\right|^2$, and a node k such that the geometric ball of squared-radius τ/n^2 around v_k contains at least cn vectors, the algorithm finds a cut $(S, \overline{S})$ with expansion at most $O(\beta n/c)$.*

PROOF. Let $d(i, j)$ denote $\left|v_i - v_j\right|^2$, and when $\{i, j\}$ is an edge e we write w_e for $d(i, j)$. The weights w_e turn the graph into a weighted graph.

Let X be the subset of nodes that correspond to the vectors in the geometric ball of radius τ/n^2 around v_k. The algorithm consists of doing a breadth-first search on the weighted graph starting from X. For $s \geq 0$ let V_s be the set of nodes whose distance from X is at most s, and let E_s be the set of edges leaving V_s. We identify s for which the cut $(V_s, \overline{V_s})$ has the lowest expansion, say α_{obs}, and output this cut.

The expansion of the cut $(V_s, \overline{V_s})$ is

$$\frac{|E_s|}{\min\left\{|V_s|, |\overline{V_s}|\right\}}.$$

Since $|V_s| \geq c \cdot |\overline{V_s}|$, the expansion is at most

$$\frac{|E_s|}{c \cdot |\overline{V_s}|},$$

allowing us to conclude $|E_s| \geq c\alpha_{obs} |\overline{V_s}|$ for all s.

To finish, we need to show that $\alpha_{obs} = O(\beta n/c)$.

Since $\sum_{i<j} d(i, j) = 1$, the triangle inequality implies that each node m also satisfies:

$$\sum_{i,j} d(i, m) + d(j, m) \geq 1,$$

which implies

$$\sum_j d(j, m) \geq 1/2n. \tag{12}$$

Separating terms corresponding to $j \in X$ and $j \notin X$ in (12) and using $c\tau < 1/8$ we obtain for the special node k:

$$\sum_{j \notin X} d(j, k) \geq \tfrac{1}{2n} - \tfrac{\tau}{n^2} \cdot cn \geq \tfrac{3}{8n}. \tag{13}$$

The lemma's hypothesis also says

$$\sum_{e \in E} w_e = \beta. \tag{14}$$

As we noticed in the proof of Corollary 2, each edge $e = \{i, j\}$ only contributes to E_s for s in the open interval (s_1, s_2), where $s_1 = d(i, X)$ and $s_2 = d(j, X)$. Triangle inequality implies that $|s_2 - s_1| \leq w_e$.

Thus

$$\beta = \sum_{e \in E} w_e \geq \int_{s>0} |E_s| \, ds \geq \int_{s>0} c\alpha_{obs} |\overline{V_s}| \, ds.$$

Furthermore, we note that

$$\int_{s>0} |\overline{V_s}| \, ds = \sum_{i \notin X} d(i, X) \geq \sum_{i \notin X} (d(i, k) - \tfrac{\tau}{n^2}).$$

Thus, from Eq. (13), we have that

$$\int_{s>0} |\overline{V_s}| \, ds \geq \tfrac{3}{8n} - n \cdot \frac{\tau}{n^2} > \frac{1}{4n}.$$

Combining the above inequalities, we obtain $\beta \geq c\alpha_{obs}/4n$, or, in other words $\alpha_{obs} = O(\beta n/c)$. □

Now, in case the conditions of Lemma 17 does not hold, we run SET-FIND.

The following lemma ensures that for any point set that is c-spread and does not meet the conditions of Lemma 17 that the procedure proceeds to Step (2) of SET-FIND with a reasonable probability for a value of σ that is $\Omega(1)$.

LEMMA 18. *Given a c-spread set of points where no more than $n/10$ of them are contained in a ball of diameter $1/10$, there is a $\sigma = O(c)$ such that Step 1 of* SET-FIND *does not HALT with probability at least $\Omega(1)$.*

PROOF. We begin with the following claim.

CLAIM 1. *For at least $1/20$ of the directions, at least $1/20$ of the pairs are separated by at least $1/90\sqrt{d}$ in the projection.*

PROOF. Consider a vertex v_i. The number of pairs v_i, v_j where $\left|v_i - v_j\right| \geq 1/10$ is at least $9(n-1)/10$. Over all pairs, the number is $9n(n-1)/10$. For unordered pairs, this is $9n(n-1)/20$.

For each such pair, the probability that v_i and v_j fall at least $1/90\sqrt{d}$ apart is at least $1/3$ from Lemma 5. Thus, the expected number of pairs that are separated by an interval of length $1/20\sqrt{d}$ is at least $9n(n-1)/60$. Using the fact that the expected number can never be larger than $n(n-1)/2$, we see that with probability at least $9/120$ at least $9n(n-1)/120$ pairs are separated by at least $1/90\sqrt{d}$. $\square$

Notice that SET-FIND with $\sigma = 1/180$ only HALTS in step 1 if more than 90% of the nodes are within $1/180\sqrt{d}$ of the median point. This does not occur with probability more than $19/20$. Thus, SET-FIND succeeds with $\Omega(1)$ probability. $\square$

To finish, we observe that if the procedure fails in Step (2), then the point set must be $(\sigma, \Theta(1), \Theta(1))$ matching covered. We can then use Lemma 15 to produce a cut or to derive a contradiction as we did before.

7. *Expander Flows: Approximate Certificates of Expansion*

Deciding whether a given graph G has expansion at least α is coNP-complete [Blum et al. 1981] and thus has no short certificate unless the polynomial hierarchy collapses. On the other hand, the value of the SDP used in the previous sections gives an "approximate" certificate of expansion. Jerrum and Sinclair [1989] and then Leighton and Rao [1999] previously showed how to use multicommodity flows to give "approximate" certificates; this technique was then clarified by Sinclair [1992] and Diaconis and Saloff-Coste [1993]. Their certificate was essentially an embedding of a complete graph into the underlying graph with minimal expansion. This certificate could certify expansion to within a $\Theta(\log n)$ factor.

The results of this section represent a continuation of that work but with a better certificate: for any graph with $\alpha(G) = \alpha$ we can exhibit a certificate to the effect that the expansion is at least $\Omega(\alpha/\sqrt{\log n})$. Furthermore, this certificate can be computed in polynomial time. The certificate involves using multicommodity flows to embed (weighted) expander graphs. The previous approaches were the same except that the expander was limited to be the complete graph. In our approach, we can choose the expander that is the easiest to embed in the underlying graph.

We remark that this view led to recent faster algorithms of Arora et al. [2004] for approximating sparsest cuts. We note that this view essentially follows from SDP duality, in the sense that the certificate we use is a feasible solution to the dual of the semi-definite program that we used in previous sections.

In this discussion, it will be more convenient to look at weighted graphs. For a weighted graph $G = (V, W)$ in which c_{ij} denotes the weight on edge $\{i, j\}$ the

sparsest cut is defined as

$$\Phi(G) = \min_{S \subseteq V : |S| \leq n/2} \frac{\sum_{i \in S, j \in \overline{S}} c_{i,j}}{|S| \, |\overline{S}|}.$$

We similarly define $\alpha(G)$.

A word on convention. Weighted graphs in this section will be symmetric, that is, $c_{ij} = c_{ji}$ for all node pairs i, j. We call $\sum_j c_{ij}$ the *degree* of node i. We say that a weighted graph is *d-regular* if all degrees are exactly d. We emphasize that d can be a fraction.

7.1. MULTICOMMODITY FLOWS AS GRAPH EMBEDDINGS.   A *multicommodity flow* in an unweighted graph $G = (V, E)$ is an assignment of a *demand* $d_{ij} \geq 0$ to each node pair i, j such that we can route d_{ij} units of flow from i to j, and can do this simultaneously for all pairs while satisfying capacity constraints. More formally, for each i, j and each path $p \in \mathcal{P}_{ij}$, there exists $f_p \geq 0$ such that

$$\forall i, j \in V \quad \sum_{p \in \mathcal{P}_{ij}} f_p = d_{ij} \tag{15}$$

$$\forall e \in E \quad \sum_{p \ni e} f_p \leq 1. \tag{16}$$

Note that every multicommodity flow in G can be viewed as an *embedding* of a weighted graph $G' = (V, E', d_{ij})$ on the same vertex set such that the weight of edge $\{i, j\}$ is d_{ij}. We assume the multicommodity flow is symmetric, that is, $d_{ij} = d_{ji}$. The following inequality is immediate from definitions, since flows do not exceed edge capacities.

$$\alpha(G) \geq \alpha(G'). \tag{17}$$

The following is one way to look at the Leighton–Rao result where K_n is the complete graph on n nodes. The embedding mentioned in the theorem is, by (17), a certificate showing that expansion is $\Omega(\alpha / \log n)$.

THEOREM 20 (LEIGHTON±RAO [1999]) *If G is any n-node graph with $\alpha(G) = \alpha$, then it is possible to embed a multicommodity flow in it with each $f_{ij} = \alpha/n \log n$ (in other words, a scaled version of K_n).*

Remark 10.   The same theorem is usually stated using all $f_{ij} = 1$ (i.e., an unweighted copy of K_n) and then bumping up the capacity of each edge of G to $O(n \log n/\alpha)$. A similar restatement is possible for our theorem about expander flows (Theorem 24).

We note that the embedding of Theorem 20 can be found in polynomial time using a multicommodity flow computation, and that this embedding is a "certificate" via (17) that $\alpha(G) = \Omega(\alpha / \log n)$.

7.2. EXPANDERS AND EIGENVALUES.   *Expanders* can be defined in more than one way. Here, we first define the *edge expansion ratio* of a cut as the ratio of the weight of edges across the cut to the total weighted degree of edges incident to the side that is smaller with respect to total weighted degree. The *edge expansion ratio* of a graph is the minimum edge expansion ratio of any cut. Furthermore, a regular weighted graph, is a weighted graph where all the nodes have the same weighted

degree where the weighted degree of a node is the sum of the weights of the incident edges. We note that the conductance is the edge expansion ratio divided by the total weighted degree of the big side of the cut.

Definition 11 (*Expanders*). For any $c > 0$, a d-regular weighted graph (w_{ij}) is a *β-expander* if for every set of nodes S, $w(S, \overline{S}) = \sum_{i \in S, j \in \overline{S}} w_{ij}$ is at least $\beta d |S|$. That is, its edge expansion ratio is at least β.

If a regular weighted graph is a β-expander, then the second eigenvalue of its Laplacian lies in the interval $[\beta^2/2, 2\beta]$ (see, e.g., Alon [1986] and Sinclair [1992].) In fact, we have the following algorithmic version from Alon[1986] as well as Alon-Milman[1985].

LEMMA 21. *There is a polynomial time algorithm that given a regular weighted graph with edge expansion ratio γ, finds a cut of weighted edge expansion ratio $\sqrt{\gamma}/2$.*

The next lemma extends the algorithm to output a *balanced* cut of small edge expansion ratio.

LEMMA 22. *For any constant $c > 0$ there is a polynomial-time algorithm that, given any regular weighted graph and a number $\gamma > 0$ behaves as follows. If the graphs has a c-balanced cut of edge expansion ratio less than γ then the algorithm outputs a $c/2$-balanced cut of edge expansion ratio $2\sqrt{\gamma}$. If the graph does not have such a cut, the algorithm finds a set of at least $(1 - c/2)n$ nodes such that the induced subgraph (with deleted edges replaced by self-loops) on them has edge expansion ratio at least 2γ.*

PROOF. First, we find a nonexpanding cut in the weighted graph using the algorithm of Lemma 21 with this value of γ. It either outputs a cut with edge expansion ratio at most $\sqrt{\gamma}$ or else fails, in which case we have verified that the graph has edge expansion ratio at least 2γ. In the former case, we delete the nodes on the small side of the cut and repeat on the remaining graph, where a self-loop is added to each node with weight equal to the removed degree for that node. We continue until the number of deleted nodes first exceeds a $c/2$ fraction of the nodes. Now there are three cases to consider.

First, if the process deletes less than $c/2$ fraction of the vertices, then the remaining graph (which has at least $(1 - c/2)n$ vertices) has edge expansion ratio 2γ, and thus in the original graph every c-balanced cut has edge expansion ratio at least γ.

Second, if the process deletes between $c/2$ and $1/2$ of the nodes, then the union of the deleted sets gives a cut with edge expansion ratio at most $\sqrt{\gamma}$ and we can therefore just output this union.

Finally, if the process deletes more than half the nodes in total then the final removed set had size greater than $c/2$ and has edge expansion ratio $2\sqrt{\gamma}$, so the algorithm can output this final removed set. □

Usually the word "expander" is reserved for the case when $\beta > 0$ is a fixed constant independent of the graph size, in which case the second eigenvalue is bigger than some fixed positive constant. Thus, an expander family can be recognized in polynomial time by an eigenvalue computation, and this motivates our definition of expander flows.

7.3. EXPANDER FLOWS.   An *expander flow* is a multicommodity flow that is a β-expander for some constant β. Such flows can be used to certify the expansion of a graph.

LEMMA 23.   *If a graph G contains a multicommodity flow (f_{ij}) that is d-regular and is a β-expander, then $\alpha(G) \geq \beta d$.*

PROOF.   For every set S of nodes, the amount of flow leaving S is at least βd, and hence this is a lower bound on the number of graph edges leaving S.   □

The previous lemma is most useful if β is constant, in which case an eigenvalue computation can be used to verify that the given flow is indeed an expander. Thus a d-regular expander flow may be viewed as a certificate that the expansion is $\Omega(d)$. The following Theorem says that such flows exist for $d = \alpha/\sqrt{\log n}$ in every graph G satisfying $\alpha(G) = \alpha$. This is an interesting structural property of every graph, in the same spirit as Theorem 20. It also yields a certificate that $\alpha(G) = \Omega(\alpha/\sqrt{\log n})$.

THEOREM 24.   *There is a constant $\beta > 0$ such that every n-vertex graph G contains a d-regular multicommodity flow in it that is a β-expander, where $d = \alpha(G)/\sqrt{\log n}$.*

The theorem above is implied by the following algorithmic version of the theorem.

THEOREM 25.   *There is a $\beta_0 > 0$ and a polynomial-time algorithm that, given a graph $G = (V, E)$ and a degree bound d, either finds a d-regular β_0-expander flow in G or else finds a cut of expansion $O(d\sqrt{\log n})$.*

The following lemma implies that to prove Theorem 25 it suffices to find a flow that expands on sets of size $\Omega(n)$ rather than all sets. It shows that given a flow that expands on large sets, one can either extend this flow to expand on all sets or find a small cut.

LEMMA 26.   *There is a polynomial time algorithm with the following property. Given a d-regular flow in a graph $G = (V, E)$ where for all $S \subset V$, $|S| \geq cn$, satisfies*

$$\sum_{i \in S, j \in \overline{S}} f_{ij} \geq \beta d |S|,$$

the algorithm either finds a cut of expansion $O(d)$ or finds a $2d$-regular $\Omega(\beta^2)$ expander flow as long as $\beta = O(1)$.

PROOF.   We apply Lemma 22 to the weighted graph induced by the flow and obtain a subgraph B (with self loops corresponding to the removed edges) which has $(1 - c/2)n \geq 3n/4$ nodes and has edge expansion ratio at least β^2.

We now augment the given flow by embedding a single flowpath to each of the deleted nodes, A, as follows. We form a flow network, with each vertex in A being the source of d units of flow and each vertex in the remaining expander to be a possible sink of d units of flow, and place capacity one on each edge. We run a max-flow computation and either route $d|A|$ units of flow or find a cut that cuts off some number say l of the sources from any remaining sink. In the latter case, the number of edges is at most dl, and thus the cut has expansion at most d.

Otherwise, the union of this flow and the previous d-regular flow can be embedded with constant congestion in G.

Furthermore, the resulting union is an $\Omega(\beta^2)$ expander flow: Any cut S where at least $1/3$ of S is in B has edge expansion ratio in the induced flow graph of $\beta^2/3$. Otherwise, at least $2/3$ of S is in A, in which case the flow sends at least $2d|S|/3$ units of flow into B at least $d|S|/3$ of which must go to nodes not in S. Thus, the cut has edge expansion ratio at least $1/3$ which is $\Omega(\beta^2)$ when $\beta = O(1)$.

To make the final expander flow regular, we can add self-loops to augment the degree of each vertex to $2d$. This changes the edge expansion ratio by at most a factor of two. $\square$

Thus, to prove Theorem 25, it suffices to prove the following lemma.

LEMMA 27. *For every $\beta > 0$, there is a $\gamma = \gamma(\beta) > 0$ and a polynomial-time algorithm that, given a graph $G = (V, E)$ and a degree bound d, either computes a d-regular flow (f_{ij}) satisfying for all $S \subset V$, $|S| \geq |V|/6$*

$$\sum_{i \in S, j \in \overline{S}} f_{ij} \geq \beta d |S|,$$

or finds a cut of expansion $\gamma d \sqrt{\log n}$.

PROOF. The algorithm tries to find a feasible solution to the following linear program, where the goal is to find a flow that expands on all large sets. Let $\mathcal{P}_{ij}$ be the set of paths that go from i to j.

$$\forall i \qquad \sum_{j} \sum_{p \in \mathcal{P}_{ij}} f_p \leq d \qquad (18)$$

$$\forall e \qquad \sum_{j} \sum_{e \in p} f_p \leq 1 \qquad (19)$$

$$\forall S \subseteq V, |V|/6 \leq |S| \leq |V|/2 \qquad \sum_{i \in S, j \in \overline{S}} \sum_{p \in \mathcal{P}_{ij}} f_p \geq \beta d |S|. \qquad (20)$$

We note that if constraint (20) was enforced for all sets rather than just large sets, the resulting linear program would specify an expander flow. Note also that we only require the degree of the demand graph to be *at most d* rather than exactly d. Any flow with max degree d can be trivially changed to a d-regular flow by adding self-loops at each node, which represent fictitious flows that use up no capacity.

Even though this linear program has exponentially many constraints, we can use the Ellipsoid algorithm to solve it approximately in the following sense. Using the algorithm of Lemma 22 as an approximate separation oracle in the Ellipsoid algorithm, we can guarantee that if the LP is feasible for a certain β, then we can find a solution that satisfies all constraints for a somewhat smaller value $\Omega(\beta^2)$. Furthermore, if the LP is infeasible for this value of β, then this algorithm finds a dual feasible solution of negative value to the closely related linear program where the constraints (20) apply to sets of size down to $|V|/6$.

Thus, the heart of the proof is to show how to produce a cut with expansion $O(d\sqrt{\log n})$ given such a dual feasible solution. We will use ideas from our earlier SDP rounding algorithm, even though we are reasoning about a linear program, not an SDP.

The dual has nonnegative variables z_S for each S of size between $|V|/6$ and $|V|/2$ and the following constraints.

$$\min \sum_e w_e + d \sum_i s_i - \beta d \sum_S z_S |S|$$

$$\sum_{e \in p} w_e + s_i + s_j \geq \sum_{S: i \in S, j \in \overline{S}} z_S \qquad \forall i, j \forall p \in \mathcal{P}_{ij}. \qquad (21)$$

$$z_S, w_e, s_i \geq 0$$

We can assume that $\sum_S z_S = 1$ by rescaling, since scaling does not affect the sign of the dual objective function.

Clearly, if the dual optimum is positive then the primal was feasible and hence an expander flow exists for that value of β. To prove Lemma 27, we show that if β is such that the dual has a negative valued solution, then we can use it to produce a cut of expansion $O(d\sqrt{\log n})$, where the constant in $O(\cdot)$ depends upon β.

Let us consider the w_e's as weights on the edges. Then, use the z_S variables to define a well-spread ℓ_2^2-representation of the graph in $\mathfrak{R}^M$, where each of the M coordinates stands for a cut S where $z_S > 0$. Each vertex x is mapped to a vector v_x where the Sth coordinate is $\pm\sqrt{z_S}$ depending on which side of S it lies. Now $|v_i - v_j|^2$ is up to a constant factor the same as the sum of z_S's for all cuts $(S, \overline{S})$ where i, j are on opposite sides. This is the so-called "cut-metric," and hence the vector representation is ℓ_2^2. (That is, $|v_i - v_j|^2$ is a valid distance function satisfying triangle inequality.) Moreover, the sum of $|v_i - v_j|^2$ over all pairs is $\Omega(n^2)$. This is because z_S is non-zero only for S's that have size $\Omega(n)$, and thus the coordinate corresponding to such a cut contributes $\Theta(z_S)$ to the ℓ_2^2 distance for a constant fraction of pairs i, j. Hence the typical pair of vertices is $\Omega(1)$ apart in ℓ_2^2 distance.

Thus, the hypothesis of Lemma 28 (the next lemma below) is satisfied. Using the algorithm of that Lemma we can either generate a cut of expansion $O(d\sqrt{\log n})$ or generate rn disjoint pairs with the following property. For each pair i, j, the length of the shortest path connecting them with respect to the weight function w_e has length at most W/nd where $W = \sum_e w_e$, and

$$|v_i - v_j|^2 = \sum_{i \in S, j \in \overline{S}} z_S = \zeta \qquad (22)$$

(i.e., the shortest path from i, j crosses a constant fraction of cuts). We show that the existence of such pairs contradicts the assumption that the dual value was negative and hence the algorithm of Lemma 28 must have generated a cut of expansion $O(d\sqrt{\log n})$.

Indeed, if such pairs exist then summing over the constraints (21) for them, we obtain

$$\frac{W}{d} + \sum_i s_i \geq r\zeta n,$$

or equivalently that

$$\sum_e w_e + d \sum_i s_i - \zeta r dn \geq 0.$$

Recall that the dual value is $\min \sum_e w_e + d \sum_i s_i - \beta d \sum_S z_S |S|$. Observing that $\sum_S z_S |S| [n/6, n]$ and choosing a sufficiently small constant β, we make the dual value positive, and this is the desired contradiction. $\square$

LEMMA 28. *For every $c > 0$ there are constants $\zeta = \zeta(c) > 0, r = r(c) > 0, \gamma = \gamma(c) > 0$ and a polynomial-time algorithm with the following property. Given a graph $G = (V, E)$, a number $d > 0$, an assignment of nonnegative weights $(w_e)_{e \in E}$ to the edges, and a ℓ_2^2-representation $v_1, v_2, \ldots, v_n$ where the distance between the average vertex pair is at least c, the algorithm either outputs a cut of capacity $\gamma d \sqrt{\log n}$ or finds a list of $r n$ node pairs (with no vertex appearing in more than one pair) such that each pair (i, j) in the list satisfies:*

$$\text{length of shortest path from } i \text{ to } j \text{ in the weighted graph} \leq \frac{W}{nd} \qquad (23)$$

$$\left| v_i - v_j \right|^2 \geq \zeta \qquad (24)$$

PROOF. Our algorithm attempts first to find a small cut as follows:

(1) Project the vectors in the ℓ_2^2-representation onto a random direction $\vec{u}$ and identify two sets A and B whose projections are $\geq m + \sigma/\sqrt{n}$ and $\leq m - \sigma/\sqrt{n}$ respectively on $\vec{u}$ where m is the median value of the projection of the vectors and σ is an appropriately small constant.

(2) It can be shown as we did in our analysis of sparsest cut that with good probability $|A|$, $|B|$ are both $\Omega(n)$. Let A be the smaller one. Solve the single commodity max flow problem in graph G where each edge has capacity $C = \lceil 1/kd \rceil$ with $k = \Theta(\sqrt{\log n})$, and there is a unit capacity edge between the source vertex and each node in A and a similar edge between the sink and each node in B. Use the algorithm to either compute a feasible flow of value $|A|$ or output the minimum cut.

Notice, if the algorithm outputs a cut for any direction, this cut has expansion at most kd. This is because this cut must disconnect l vertices in A (respectively, B) from the sink (respectively, source), and the capacity of this cut must be at most l. Hence, the number of edges in the cut must be at most $l/C \leq lkd$. Thus, in this case, we obtain a cut of expansion at most kd.

So suppose the algorithm finds a flow of value $|A|$ for most directions u. We show how to apply Lemma 15 to either find a cut of expansion $O(d\sqrt{\log n})$ or to produce a pair (i, j) satisfying conditions (23) and (24). (We then indicate how the argument actually yields $\Omega(n)$ such pairs.) To apply that lemma, we must form matching covers and define a proximity graph. Below, *distance* will refer to distance in the original weighted graph with respect to the weights w_e's on the edges, whereas *geometric* distance refers to squared Euclidean distance among the corresponding vectors.

The definition of the proximity graph is easy: there is an edge between each pair of vertices (i, j) whose distance most W/kdn. If a pair of vertices is adjacent in the proximity graph and also has geometric distance $\Omega(1)$, then such a pair satisfies (23) and (24) and we are done. So assume that every pair of adjacent vertices in the proximity graph have geometric distance at most ζ for an arbitrarily small $\zeta > 0$.

Now we define the matching covers. For each direction $\vec{u}$, use flow decomposition on the flow computed in Step (2) above to obtain $\Omega(n)$ flowpaths. These flowpaths

naturally define a pairing of nodes, since each flowpath consists of a source edge to some node $a \in A$ followed by a path to node $b \in B$ followed by a sink edge from b to the sink. Such a flowpath can be thought of as pairing a with b, and since each of these nodes has a unique edge to the source/sink, this pairing is one-to-one (i.e., a partial matching). Now define the *length* of the flowpath to be the sum of the edge weights in the path. The total length of all the flowpaths is at most CW since each edge can appear in at most C flowpaths. Thus, the average flowpath has length at most $CW/|A| = O(W/nkd)$. For a direction $\vec{u}$, retain only those pairs in the partial matching where the flowpath length is at most twice the average length. This yields a matching with $\Omega(n)$ pairs where the pairs are within $O(W/nkd)$ distance.

Now we have the matching covered point set and the ζ-proximate graph as required by the hypothesis of Lemma 15. Let $k' = \Theta(\sqrt{\log n})$ be some large enough multiple of $\sqrt{\log n}$, such that as in our earlier proofs, it is impossible for the set $S_{k'}$ in the lemma statement to be nonempty (simply because no element can be projection covered with such large projections). Then, either Case (2) or (3) of Lemma 15 must hold for k'.

If case 3 of Lemma 15 holds, then we obtain a nonmagnifying set S of size $\Omega(n)$ in the ζ-proximate graph. Hence, the set of nodes with no direct edges to S in the proximate graph (i.e., nodes whose distance to S in the original graph is at most W/kdn) have size $\Omega(n)$. Now we choose a random positive δ in the interval $[0, W/kdn]$ and output the set containing all nodes whose distance (with respect to the weights w_e's on the edges) to the closest point of S is at most δ. The expected number of edges in this cut is at most $W/(W/kdn) = kdn$. As already noted, each side of the cut has $\Omega(n)$ nodes. Thus, the expected expansion of the cut is $O(kd)$.

Case (2) of Lemma 15 yields a pair of vertices i, j whose geometric distance is $\Omega(1)$ and that are k' matching hops away, which corresponds to distance $k'W/kdn = O(W/dn)$ in the original graph since both k' and k are $\Theta(\sqrt{\log n})$.

Thus, in every case, we obtain a pair (i, j) satisfying (23) and (24). Now we note that the argument can be repeated $\Omega(n)$ times to generate more such pairs. Indeed, deleting this pair leaves a set of vertices that remains matching covered for slightly worse parameters; it removes at most 2 pairs from the $\Theta(n)$ pairs in each matching. Thus, either the process can be repeated $\Theta(n)$ times to yield $\Theta(n)$ pairs or it fails at some point and produces a cut in some iteration as stated in the lemma. □

We note that as before the analysis can be simplified using Lee's [2005] techniques. In particular, cuts can be produced using only a projection based procedure as is the case for finding sparse cuts using semidefinite programming.

7.4. EXPANDER FLOWS AS DUAL SOLUTIONS.   Now we show that expander flows are actually a natural family of dual solutions to the SDP for SPARSEST CUT. Denote by $OPT(W)$ the value of the simplest SDP relaxation (with no triangle inequality) of this problem: find vectors $v_1, v_2, \ldots, v_n \in \Re^n$ so as to:

$$\min \sum_{\{i,j\}} c_{i,j} \left| v_i - v_j \right|^2 \tag{25}$$

$$\sum_{i \neq j} \left| v_i - v_j \right|^2 = 1 \tag{26}$$

Let $OPT_\Delta(W)$ denote the value of the relaxation when we also impose the triangle inequality constraints. Now we relate $OPT(\cdot)$ and $OPT_\Delta(\cdot)$.

Recall from our discussion of expander flows that a multicommodity flow in G can be viewed as an *embedding* of the demand graph $D = (d_{ij})$. We will show that such a flow is a dual solution to the SDP relaxation with triangle inequality (we have learnt that this observation was known to others, including M. Sudan and D. Williamson, personal communication).

LEMMA 29 (FLOWS AS DUAL SOLUTIONS) *If a demand graph $D = (d_{ij})$ can be embedded in $G = (V, W)$, then*

$$OPT_\Delta(W) \geq OPT(D).$$

Remark 11. Of course, when D is an expander, then the value $OPT(D)$ is (upto scaling) essentially the same as the eigenvalue bound for sparsest cut.

To prove the lemma, we need to write the duals. Let us do this for the simple SDP, first rewriting it using x_{ij} and z_{ij} to replace $|v_i - v_j|^2$ and $< v_i, v_j >$, respectively. We use $\text{diag}(a_i)$ to denote the $n \times n$ matrix which has $a_1, a_2, \ldots, a_n$ on the diagonal and 0 everywhere else.

PRIMAL	*DUAL*	(27)
$\min \sum_{\{i,j\}} c_{ij} x_{ij}$	$\max \quad z$	(28)
$\sum_{i,j} x_{ij} = 1$	$z + s_{ij} = c_{ij} \quad \forall i, j$	(29)
$\forall i, j \quad x_{ij} + 2z_{ij} - z_{ii} - z_{jj} = 0$	s_{ij} unrestricted	(30)
(z_{ij}) is psd	$\text{diag}\left(\sum_{j \neq i} s_{ij}\right) - (s_{ij})_{i \neq j}$ is psd	
		(31)

Now we consider the SDP with triangle inequality. The usual formulation is to write for each triple i, j, k an additional constraint: $x_{ij} + x_{jk} \geq x_{ik}$. We use an equivalent but more complicated formulation: if $\mathcal{P}_{ij}$ denotes the set of paths from i to j in the *complete* graph on n vertices, then

$$\forall i, j, \forall p \in \mathcal{P}_{ij}, \quad \sum_{(u,v) \in p} x_{uv} \geq x_{ij}. \tag{32}$$

Now let us write the primal and the dual.

PRIMAL	*DUAL*	
$\min \sum_{\{i,j\}} c_{ij} x_{ij}$	$\max \quad z$	
$\sum_{i,j} x_{ij} = 1$	$z + s_{ij} - \sum_{p \in \mathcal{P}_{ij}} f_p + t_{ij} = c_{ij} \quad \forall i, j$	(33)

$$\forall i, j \quad x_{ij} + 2z_{ij} - z_{ii} - z_{jj} = 0 \quad s_{ij} \text{ unrestricted} \tag{34}$$

$$\forall i, j, \forall p \in \mathcal{P}_{ij}, \quad \sum_{(u,v)\in p} x_{uv} \geq x_{ij} \tag{35}$$

$$(z_{ij}) \text{ is psd} \qquad\qquad \mathrm{diag}(\sum_{j\neq i} s_{ij}) - (s_{ij})_{i\neq j} \text{ is psd} \tag{36}$$

Here t_{ij} is a shorthand for $\sum_{p\ni\{i,j\}} f_p$.

Now let D be the demand graph of any multicommodity flow that can be embedded in G. A flow is an assignment of nonnegative values to f_p's such that $t_{ij} \leq c_{ij}$ (indeed, the only nonzero f_p's are for p that are actually paths in G). Fixing these values of f_p's, finding the best values to z and the s_{ij}'s is nothing but the same as finding $OPT(W')$ where W' is the weighted graph in which W'_{ij} is $c_{ij} - t_{ij} + D_{ij}$, where D_{ij} is the total flow from i to j, namely $\sum_{p\in\mathcal{P}_{ij}} f_p$. Of course, since $t_{ij} \leq c_{ij}$, this is lower bounded by $OPT(D)$ where D is the demand graph of the flow. Thus the lemma is proved.

8. *The Open Questions*

Currently, it is completely plausible that our techniques can be strengthened to achieve an approximation ratio much better than $\sqrt{\log n}$ for UNIFORM SPARSEST CUT. Originally it was suggested in Goemans [1998] that the SDP with triangle inequality has an integrality gap of $O(1)$ but Khot and Vishnoi [2005] showed that the gap is $\Omega(\log\log^{1/6} n)$, which was improved to $\Omega(\log\log n)$ by Krauthgamer and Rabani [2006] for the nonuniform sparsest cut. Recently, Devanur et al. [2006] showed that the $\Omega(\log\log n)$ gap holds even for the uniform version. As noted earlier, the example of the Boolean hypercube shows that our main theorem (Theorem 1) about the existence of large Δ-separated subsets cannot be improved, so some newer idea seems necessary.

Below we provide a suggested roadmap for attacking these questions. We three questions. All concern any constant degree graph $G = (V, E)$ and any unit-ℓ_2^2-representation $v_1, v_2, \ldots, v_n$, that is c-spread for some constant c.

Is there a function $s(n) = o(\sqrt{\log n})$ such that:

Question 1. there is always an edge $\{i, j\} \in E$ such that $|v_i - v_j|^2 = \Omega(\alpha/s(n))$?

Question 2. there are pairs of vertices i, j such that $d_G(i, j) = O(s(n)/\alpha)$ and $|v_i - v_j|^2 = \Omega(1)$?

Question 3. Version of Question 2 whereby for some $\tau > 0$, an arbitrary set of τ fraction of nodes are forbidden from being chosen as i, j and nevertheless these i, j exist.

Earlier in this article, we proved that the answer to these questions is Yes if $s(n) = O(\sqrt{\log n})$. The authors could not agree on a guess as to whether or not the answer is Yes to any of these questions for an $s(n)$ that grows slower.

Now we summarize, without detailed proofs, the implications if the answers to these questions are Yes.

LEMMA 30

(1) *If the integrality gap of the SDP is $O(s(n))$, then the answer to Question 1 is Yes.*

(2) *A Yes answer to Question* 2 *suffices to prove that the integrality gap is* $O(s(n))$.
(3) *A Yes answer to Question* 3 *implies the existence of expander flows in graphs that certify expansion of* $\alpha/s(n)$. *Namely, Theorem 24 is true with* $d = \alpha/s(n)$.
(4) *A Yes answer to Question* 2 *implies YES for Question* 1.

PROOF.

(1) If the integrality gap of the SDP is $O(s(n))$, then the value of the objective is $\Omega(\alpha n/s(n))$ and hence at least one edge has value $\Omega(\alpha/s(n))$.
(2) Uses a modification of our original proof that the integrality gap is $O(\sqrt{\log n})$.
(3) Question 3 is a stronger version of statements proved earlier. One can mimic the proof of Theorem 24 (specifically, the last few lines of Lemma 27) using this stronger version. □

8.1. BRINGING THE QUESTION ™DOWN TO ℓ_1. Before explaining the title of this section we first recall the connection between cut problems and embeddings into ℓ_1. A unit ℓ_2^2-representation $v_1, v_2, \ldots, v_n$ of a graph is said to be ℓ_1 if there is a set of vectors $u_1, u_2, \ldots, u_n$ such that $\left|v_i - v_j\right|^2 = \left|u_i - u_j\right|_1$. We say that it is ℓ_1 *upto distortion* c if $\left|u_i - u_j\right|_1 \le \left|v_i - v_j\right|^2 \le c \cdot \left|u_i - u_j\right|_1$. The importance of embeddings to cut problems arises from the following equivalent characterization.

LEMMA 31 (WELL-KNOWN) *A unit* ℓ_2^2-*representation* $v_1, v_2, \ldots, v_n$ *is* ℓ_1 *iff there is a* $\alpha_S \ge 0$ *associated with each cut* $(S, \overline{S})$ *such that*

$$\left|v_i - v_j\right|^2 = \sum_S \alpha_S d_S(i, j), \tag{37}$$

where $d_S(i, j) = 0$ *if* i, j *are on the same side of the cut and* 4 *otherwise. (In other words,* ℓ_1 *representations correspond exactly to the cut cone.)*

PROOF. One direction is trivial from convexity since every cut metric is ℓ_1.
For a proof of the other direction see Shmoys' survey [1995] or Matousek's book [2002]. □

Thus, minimization over ℓ_1 metrics is exactly equivalent to minimizing over cuts (and thus **NP**-hard). Goemans and Linial independently conjectured that every ℓ_2^2-representation is ℓ_1 upto a distortion factor $O(1)$. If true, this would have implied an integrality gap of $O(1)$ for NONUNIFORM SPARSEST CUT, but recently Khot and Vishnoi [2005] showed that distortion $(\log \log n)^\epsilon$ is necessary. However, the best upper bound on the distortion is $O(\sqrt{\log n} \log \log n)$, so a large gap remains.
One may reasonably question whether it makes much sense to focus attention exclusively on ℓ_2^2 metrics when these metrics merely happen to capture the properties of *one* family of SDPs. Shouldn't one expend more effort in identifying stronger SDP relaxations by specifying more constraints than the triangle inequality? After all, procedures such as Lovász–Schrijver [1991] give us an infinite hierarchy of progressively tighter SDP relaxations to choose from. (The triangle inequality constraint is implied already in the relaxation at the third level of this hierarchy.)
Now we point out that at least in context of our Questions, adding other constraints (in addition to the triangle inequality) to the SDP may not help. We bring "down" the Questions to ℓ_1 so that it suffices to prove them for ℓ_1 metrics. Thus although adding other constraints could conceivably give a class of metrics that is a proper

subset of ℓ_2^2, proving our conjectures for that other class of metrics will not be any easier, since that class of metrics (being a relaxation of cut metrics) would always contain ℓ_1.

[**Question 4:**] If the unit-ℓ_2^2-representation is actually ℓ_1, then there is a vertex pair i, j such that $d_G(i, j) = O(s(n)/\alpha)$ and $|v_i - v_j|^2 = \Omega(1)$.

LEMMA 32. *Questions 2 and 4 are equivalent.*

PROOF. Yes to Question 2 $\Rightarrow$ Yes to Question 4: Since ℓ_1 metrics are a subcase of ℓ_2^2 metrics, Question 4 is trivially a subcase of Question 2.

Yes to Question 4 $\Rightarrow$ Yes to Question 2: Given any unit-ℓ_2^2-representation $v_1, v_2, \ldots, v_n$ of the graph, consider the uniform distribution on all hyperplane cuts (a la Goemans and Williamson [1995]). Represent this distribution using Lemma 31 by an ℓ_2^2-representation that is ℓ_1, namely, a set of unit vectors $u_1, u_2, \ldots, u_n$ where

$$|u_i - u_j|^2 = \Pr[i, j \text{ are separated in hyperplane cut produced using } v_1, \ldots, v_n].$$

Let i, j be the pair of nodes in G whose existence follows if Question 4 has a Yes answer. Namely, $|u_i - u_j|^2 = \Omega(1)$. Then, these also suffice for Question 2 since if

$$\Pr[i, j \text{ are separated in hyperplane cut on } v_1, \ldots, v_n] = \Omega(1),$$

then $|v_i - v_j|^2 = \Omega(1)$ also.   □

Remark 12. We can make a robust form of Question 4 whereby the pair exists even if we forbid τn nodes from playing the role of i, j. Giving a Yes answer to that question is *equivalent* to showing that expander flows can be used to certify an expansion of $\alpha/s(n)$.

9. *Conclusions*

In the four years since the initial distribution of this article, the geometric techniques introduced here have found many uses in designing other SDP-based approximation algorithms as well as resolving open questions in metric space embeddings. We see several promising avenues of research in the near future.

First, there appears to be plenty of scope to use more sophisticated geometric arguments in analyzing SDPs. Procedures such as Lovász–Schrijver [1991] or Sherali–Adams [1990] in principle give an infinite family of SDP relaxations that have thus far resisted all attempts at analysis but may have low integrality gaps. Since the triangle inequality constraints are obtained trivially in these frameworks, perhaps our analysis of the triangle inequality constraints gives a hint how to analyze more complicated SDPs.

Second, as already mentioned even the SDPs with triangle inequality are not fully understood as yet: there is a big gap between the lower bounds and upper bounds known on the integrality gap. A similar observation holds for questions about geometric embeddings such as the distortion required to embed ℓ_2^2 metrics into ℓ_1.

Finally, there is a need to design efficient approximation algorithms, since solving SDPs is fairly inefficient (though polynomial time). Solving the SDP of (4)–(7) takes about $n^{4.5}$ time for an n-node graph using interior point methods. As mentioned, a more efficient $\tilde{O}(n^2)$-time algorithm is now known [Arora et al. 2004]

that uses expander flows. Recently a more general *primal-dual* framework was invented [Arora and Kale 2007] for replacing SDPs with faster approximation algorithms for problems other than SPARSEST CUT. Our Structure Theorem plays an important role in the analysis of the running time. It would be interesting to continue to improve the running times, and to extend the framework to even more problems.

Finally, we feel expander flows ought to have other applications besides estimating the edge expansion ratio/expansion. Some applications to geometric analysis have already been found [Naor et al. 2005].

REFERENCES

AGARWAL, A., CHARIKAR, M., MAKARYCHEV, K., AND MAKARYCHEV, Y. 2005. $o(\sqrt{\log n})$ approximation algorithms for min uncut, min 2cnf deletion, and directed cut problems. In *STOC '05: Proceedings of the 34th annual ACM Symposium on Theory of Computing*. ACM, New York, 573–581.

ALIZADEH, F. 1995. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM J. Optim. 5*, 1, 13–51.

ALON, N. 1986. Eigenvalues and expanders. *Combinatorica 6*, 2, 83–96.

ALON, N., AND MILMAN, V. D. 1985. λ_1, isoperimetric inequalities for graphs, and superconcentrators. *J. Combin. Theory Ser. B 38*, 1, 73–88.

ARORA, S., HAZAN, E., AND KALE, S. 2004. $0(\sqrt{\log n})$ approximation to sparsest cut in $\tilde{O}(n^2)$ time. In *FOCS '04: Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science (FOCS'04)*. IEEE Computer Society Press, Los Alamitos, CA, 238–247.

ARORA, S., AND KALE, S. 2007. A combinatorial, primal-dual approach to semidefinite programs. In *STOC '07: Proceedings of the 39th Annual ACM Symposium on Theory of Computing*. ACM, New York, 227–236.

ARORA, S., LEE, J. R., AND NAOR, A. 2008. Euclidean distortion and the sparsest cut. *J. Amer. Math. Soc. 21*, 1, 1–21 (electronic).

AUMANN, Y., AND RABANI, Y. 1998. An $O(\log k)$ approximate min-cut max-flow theorem and approximation algorithm. *SIAM J. Comput. 27*, 1, 291–301 (electronic).

BALL, K. 1997. An elementary introduction to modern convex geometry. In *Flavors of Geometry*. Mathematics Science Research Institute Publisher, vol. 31. Cambridge Univ. Press, Cambridge, 1–58.

BLUM, M., KARP, R., VORNBERGER, O., PAPADIMITRIOU, C., AND YANNAKAKIS, M. 1981. The complexity of testing whether a graph is a superconcentrator. *Inf. Proc. Letters 13*, 164–167.

CHARIKAR, M., HAJIAGHAYI, M. T., KARLOFF, H., AND RAO, S. 2006. l22 spreading metrics for vertex ordering problems. In *SODA '06: Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithm*. ACM, New York, 1018–1027.

CHAWLA, S., GUPTA, A., AND RÄCKE, H. 2005. Embeddings of negative-type metrics and an improved approximation to generalized sparsest cut. In *SODA '05: Proceedings of the 16th annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, Philadelphia, PA, 102–111.

CHEEGER, J. 1970. A lower bound for the smallest eigenvalue of the Laplacian. *Probl. Analysis*, 195–199.

CHUNG, F. R. K. 1997. *Spectral graph theory*. CBMS Regional Conference Series in Mathematics, vol. 92. Published for the Conference Board of the Mathematical Sciences, Washington, DC.

DANZER, L., AND BRANKO, G. 1962. On two problems of P. Erdos and V. L. Klee concerning convex bodies *(in German)*. *Math. Zeitschrift 79*, 95–99.

DEVANUR, N., KHOT, S., SAKET, R., AND VISHNOI, N. 2006. Integrality gaps for sparsest cut and minimum linear arrangement problems. In *STOC'06: Proceedings of the 38th Annual ACM Symposium on Theory of Computing*. ACM, New York, 537–546.

DIACONIS, P., AND SALOFF-COSTE, L. 1993. Comparison theorems for reversible markov chains. *Ann. Appl. Prob. 3*, 696–730.

ENFLO, P. 1969. On the nonexistence of uniform homeomorphisms between L_p-spaces. *Ark. Mat. 8*, 103–105.

FEIGE, U., HAJIAGHAYI, M., AND LEE, J. R. 2005. Improved approximation algorithms for minimum-weight vertex separators. In *STOC '05: Proceedings of the 37th Annual ACM Symposium on Theory of Computing*. ACM, New York, 563–572.

FEIGE, U., AND LEE, J. R. 2007. An improved approximation ratio for the minimum linear arrangement problem. *Inform. Process. Lett. 101*, 1, 26–29.

GOEMANS, M. X. 1998. Semidefinite programming and combinatorial optimization. In *Proceedings of the International Congress of Mathematicians*, Vol. III (Berlin, 1998). *Doc. Math*., 657–666 (electronic).

GOEMANS, M. X., AND WILLIAMSON, D. P. 1995. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM 42*, 6, 1115–1145.

GRÖTSCHEL, M., LOVÁSZ, L., AND SCHRIJVER, A. 1993. *Geometric algorithms and combinatorial optimization*, Second ed. Algorithms and Combinatorics, vol. 2. Springer-Verlag, Berlin, Germany.

JERRUM, M., AND SINCLAIR, A. 1989. Approximating the permanent. *SIAM J. Comput. 18*, 6, 1149–1178.

KARAKOSTAS, G. 2005. A better approximation ratio for the vertex cover problem. In *Automata, languages and programming*. Lecture Notes in Computer Science vol. 3580. Springer-Verlag, Berlin, Germany, 1043–1050.

KARGER, D., MOTWANI, R., AND SUDAN, M. 1998. Approximate graph coloring by semidefinite programming. *J. ACM 45*, 2, 246–265.

KARLOFF, H. J., AND ZWICK, U. 1997. A 7/8-approximation algorithm for max 3sat? In *Proceedings of the 38th IEEE Foundations of Computer Science (FOCS)*. IEEE Computer Society Press, Los Alamitos, CA, 406–415.

KHANDEKAR, R., RAO, S., AND VAZIRANI, U. 2006. Graph partitioning using single commodity flows. In *STOC '06: Proceedings of the 38th Annual ACM Symposium on Theory of Computing*. ACM, New York, 385–390.

KHOT, S., AND VISHNOI, N. 2005. The unique games conjecture, integrality gap for cut problems and embeddability of negative type metrics into ℓ1. In *FOCS '05: Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*. IEEE Computer Society Press, Los Alamitos, CA, 53–62.

KRAUTHGAMER, R., LEE, J. R., MENDEL, M., AND NAOR, A. 2005. Measured descent: a new embedding method for finite metrics. *Geom. Funct. Anal. 15*, 4, 839–858.

KRAUTHGAMER, R., AND RABANI, Y. 2006. Improved lower bounds for embeddings into l1. In *SODA '06: Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithm*. ACM, New York, 1010–1017.

LANG, K., AND RAO, S. 2004. A flow-based method for improving the expansion or conductance of graph cuts. In *Proceedings of the 10th International IPCO Conference*. Lecture Notes in Computer Science, vol. 3064. Springer-Verlag, Berlin, Germany, 325–337.

LEE, J. R. 2005. On distance scales, embeddings, and efficient relaxations of the cut cone. In *SODA '05: Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, Philadelphia, PA, USA, 92–101.

LEIGHTON, T., AND RAO, S. 1999. Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms. *JACM 46*, 6, 787–832.

LINIAL, N., LONDON, E., AND RABINOVICH, Y. 1995. The geometry of graphs and some of its algorithmic applications. *Combinatorica 15*, 2, 215–245.

LOVÁSZ, L., AND SCHRIJVER, A. 1991. Cones of matrices and set-functions and 0-1 optimization. *SIAM J. Optim. 1*, 2, 166–190.

MATOUŠEK, J. 2002. *Lectures on Discrete Geometry*. Graduate Texts in Mathematics, vol. 212. Springer-Verlag, New York.

NAOR, A., RABANI, Y., AND SINCLAIR, A. 2005. Quasisymmetric embeddings, the observable diameter, and expansion properties of graphs. *J. Funct. Anal. 227*, 2, 273–303.

NESTEROV, Y., AND NEMIROVSKII, A. 1994. *Interior-point polynomial algorithms in convex programming*. SIAM Studies in Applied Mathematics, vol. 13. SIAM, Philadelphia, PA.

SCHECHTMAN, G. 2003. Concentration, results and applications. In *Handbook of the Geometry of Banach Spaces, volume 2*, W. Johnson and J. Lindenstrauss, Eds. North Holland, Amsterdam, The Netherlands, (Draft version available from Schechtman's website).

SHAHROKHI, F., AND MATULA, D. W. 1990. The maximum concurrent flow problem. *J. Assoc. Comput. Mach. 37*, 2, 318–334.

SHERALI, H., AND ADAMS, W. 1990. A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. *SIAM J. Discrete Math. 3*, 3, 411–430.

SHMOYS, D. S. 1995. Cut problems and their application to divide and conquer. In *Approximation Algorithms for NP-Hard Problems*, D. Hochbaum, Ed. PWS Publishing.

SINCLAIR, A. 1992. Improved bounds for mixing rates of Markov chains and multicommodity flow. *Combinat. Probab. Comput. 1*, 4, 351–370.