Breaking the Multicommodity Flow Barrier for $O(\sqrt{\log n})$-Approximations to Sparsest Cut

Jonah Sherman*
University of California, Berkeley

August 10, 2009

Abstract

This paper ties the line of work on algorithms that find an $O(\sqrt{\log n})$-approximation to the SPARSEST CUT together with the line of work on algorithms that run in sub-quadratic time by using only single-commodity flows. We present an algorithm that simultaneously achieves both goals, finding an $O(\sqrt{\log(n)/\varepsilon})$-approximation using $O(n^{\varepsilon}\log^{O(1)} n)$ max-flows. The core of the algorithm is a stronger, algorithmic version of Arora *et al.*'s structure theorem, where we show that matching-chaining argument at the heart of their proof can be viewed as an algorithm that finds good augmenting paths in certain geometric multicommodity flow networks. By using that specialized algorithm in place of a black-box solver, we are able to solve those instances much more efficiently.

We also show the cut-matching game framework can not achieve an approximation any better than $\Omega(\log(n)/\log\log(n))$ without re-routing flow.

1 Introduction

We consider the problem of partitioning a graph into relatively independent pieces in the sense that not too many edges cross between them. Two concrete optimization problems arising in that context are the SPARSEST CUT and BALANCED SEPARATOR problems. We are given an undirected weighted graph G on n vertices, where each edge xy has capacity G_{xy} (we identify a graph with its adjacency matrix). The *edge expansion* of a cut $(S, \overline{S})$ is $h(S) = \frac{\sum_{x\in S, y\in\overline{S}} G_{xy}}{\min\{|S|,|\overline{S}|\}}$. The SPARSEST CUT problem is to find a cut $(S, \overline{S})$ minimizing $h(S)$; we write $h(G)$ to denote the value of such a cut. The BALANCED SEPARATOR problem has the same objective but the additional constraint that $\min\{|S|, |\overline{S}|\} \geq \Omega(n)$. Both problems are NP-hard, so we settle for approximation algorithms.

Most of the original work on graph partitioning focused on achieving the best approximation factor and falls into one of two themes. The first is based on multicommodity flow, using the fact that if a graph H of known expansion can be routed in G via a feasible flow, then $h(H) \leq h(G)$. If H is some fixed graph, finding the best possible lower bound is equivalent to solving the maximum concurrent flow problem; i.e., maximizing α such that $F \leq G$ and $D \geq \alpha H$ where $D_{xy} = \sum_{p:x\leftrightarrow y} f_p$ is the *demand graph* and $F_{xy} = \sum_{p\ni xy} f_p$ is the *flow graph* of the underlying flow. By taking H to be the complete graph, Leighton and Rao showed an upper bound of $h(G) \leq O(\log n)\alpha^* h(H)$ for the optimal α^*, yielding an $O(\log n)$ approximation. The other theme is the discrete Cheeger's inequality of Alon and Milman[2] characterizing the relationship between cuts and the spectrum of a graph's Laplacian matrix. In particular, if G has maximum degree d, then $\lambda_2(\mathcal{L}_G)/2 \leq h(G) \leq \sqrt{2d\lambda_2(\mathcal{L}_G)}$, where $\lambda_2(\mathcal{L}_G)$ is second smallest eigenvalue of G's Laplacian. The two themes are incomparable, as the latter is a better approximation when G is an expander (i.e., $h(G)/d$ is large) while the former is better when G has sparse cuts.

Arora, Rao, and Vazirani naturally combined the two themes. Rather than embedding a fixed graph H of known expansion, they embed an arbitrary H and then certify H's expansion via $\lambda_2(\mathcal{L}_H)$[7]. Since $\lambda_2(\mathcal{L}_H) \geq \alpha$ is equivalent to $\mathcal{L}_H \succeq \frac{\alpha}{n}\mathcal{L}_K$, where K is the complete graph, the problem of finding the best such lower-bound can be cast as a semidefinite program:

$$\max \alpha \qquad s.t. \quad \frac{\alpha}{n}\mathcal{L}_K \preceq \mathcal{L}_D, \quad F \leq G \tag{1}$$

*Research supported by a UC Berkeley Regents Fellowship and in part by NSF grant CCF-0635401

They showed that for the optimal α^*, one has an upper bound of $h(G) \leq O(\sqrt{\log n})\alpha^*$, yielding the currently best known approximation factor. Shortly thereafter, Arora, Hazan, and Kale designed a primal-dual algorithm to approximately solve (1) in $\tilde{O}(n^2)$ time using multicommodity flows[3].

More recently, researchers have focused on designing efficient algorithms for graph partitioning that beat the quadratic multicommodity flow barrier. Khandekar, Rao, and Vazirani designed a simple primal-dual framework for constructing such algorithms based on the *cut-matching game* and showed one could achieve an $O(\log^2 n)$ approximation in that framework using polylog max-flows[15]. Arora and Kale designed a very general primal-dual framework for approximately solving SDPs[5]. They showed efficient algorithms for several problems could be designed in their framework, including an $O(\log n)$-approximation to SPARSEST CUT using polylog max-flows. They also showed one could achieve an $O(\sqrt{\log n})$-approximation in their framework using multicommodity flows, simplifying the previous algorithm of [3]. Orecchia *et al.* extended the cut-matching game framework of [15] to achieve an $O(\log n)$ approximation[20]. They present two slightly different algorithms, and remarkably, their second algorithm is the same as Arora and Kale's, even though they never explicitly mention any SDP. They also showed a lower bound of $\Omega(\sqrt{\log n})$ on the approximation factor achievable in the cut-matching framework, suggesting the framework might precisely capture the limits of current approximation algorithms and posed the question of whether $O(\sqrt{\log n})$ could be efficiently achieved in that framework.

1.1 This Paper.

We tie those two lines of work together by simultaneously achieving the $O(\sqrt{\log n})$ approximation factors of the former with the nearly max-flow running time of the latter.

Theorem 1.1. *For any $\varepsilon \in [O(1/\log(n)), \Omega(1)]$, there is an algorithm to approximate the* SPARSEST CUT *and* BALANCED SEPARATOR *problems to within a factor of $O(\sqrt{\log(n)/\varepsilon})$ using only $O\left(n^{\varepsilon} \log^{O(1)}(n)\right)$ max-flows.*

Theorem 1.1 effectively subsumes the results of [3, 15, 5, 20], as taking $\varepsilon = \Theta(1/\log(n))$ yields an $O(\log(n))$ approximation using polylog max-flows, while any constant $\varepsilon < 1/2$ achieves an $O(\sqrt{\log(n)})$ approximation in sub-quadratic $\tilde{O}(m + n^{3/2+\varepsilon})$ time using the max-flow algorithm of Goldberg and Rao[14]. We also show the cut-matching game framework of [15] can not achieve an approximation better than $\Omega(\log(n)/\log\log(n))$ without re-routing flow.

We build heavily on Arora and Kale's work, achieving our improvement by replacing their use of a black-box multicommodity flow solver with a specialized one that makes use of the additional structure present in the flow instances that arise. We begin in section 2 by reviewing the nature of those flow problems, as well as the main ideas behind the algorithms of [3, 5, 20]. Having clarified the connection to partitioning, we also state our main technical result, theorem 2.3. In section 3 we describe the details of our algorithm, the correctness of which follows immediately from theorem 2.3. The proof of theorem 2.3 appears in section 4. Our lower-bound for the cut-matching game is then discussed in 5, and we finish with some concluding remarks in section 6.

2 Expander Flows

Expander-flow based algorithms all work by approximately solving (1), either explicitly as in [3, 5], or implicitly as in [15, 20], by iteratively simulating play of its corresponding two-player zero-sum game. The game has two players: the embedding player and the flow player. The embedding player chooses a non-trivial embedding $V = (\mathbf{v}_1, \ldots, \mathbf{v}_n) \in (\mathbb{R}^d)^n$ of the vertices of G. The flow player chooses a feasible flow $F \leq G$ supporting demands D with the goal of routing flow between points that are far away in the embedding. More precisely, the payoff to the flow player is:

$$\Phi(V, D) = \frac{\sum_{x<y} D_{xy} \|\mathbf{v}_x - \mathbf{v}_y\|^2}{\frac{1}{n}\sum_{x<y} \|\mathbf{v}_x - \mathbf{v}_y\|^2}$$

For given demands D, the best response for the embedding player is the one-dimensional embedding given by an eigenvector of $\mathcal{L}_D$ of eigenvalue $\lambda_2(\mathcal{L}_D)$, yielding a value of $\lambda_2(\mathcal{L}_D)$. On the other hand, for a given embedding, the best response for the flow player is a solution to the weighted maximum multicommodity flow problem given by

$$\max \sum_{x<y} D_{xy} \|\mathbf{v}_x - \mathbf{v}_y\|^2 \qquad s.t. \quad F \leq G \tag{2}$$

The frameworks of [15, 5, 20] start with an initial embedding V^1, such as all points roughly equidistant. On a given iteration t, the algorithm presents V^t to the flow player, who must either respond with demands D^t of value $\Phi(V^t, D^t) \geq 1$, or a cut C^t of expansion at most κ, where κ is the desired approximation factor. In the latter case the algorithm terminates; in the former case, the demands are used to update the embedding for the next iteration. The precise update differs among each algorithm, but essentially vertices x, y with large D^t_{xy} will be squeezed together in the embedding. The analysis of [15, 5, 20] show that after T iterations, for sufficiently large T, their adaptive strategies actually played nearly as well as they could have in hindsight, in that

$$\lambda_2\left(\frac{\mathcal{L}_{D^1} + \cdots + \mathcal{L}_{D^T}}{T}\right) \geq \Omega(1)$$

Since averaging T feasible flows yields a feasible flow, after T iterations the graph $D = (D^1 + \cdots + D^T)/T$ with $\lambda_2(\mathcal{L}_D) \geq \Omega(1)$ has been routed in G. Thus, for a given graph, the algorithm either routes an $\Omega(1)$-expander-flow in G or else finds a cut of expansion κ. Using a binary search and scaling the edge capacities appropriately yields an $O(\kappa)$ approximation algorithm.

The embedding can be updated in nearly linear time, and $T = O(\log^{O(1)}(n))$, so the running time of such algorithms is dominated by the running time of the flow player. By sparsifying G (using e.g. [8]), we can and shall assume it has $m = O(n \log n)$ edges. Using Fleischer's multicommodity flow algorithm[12] as a black box, a nearly optimal pair of primal/dual solutions to (2) can be computed in $\tilde{O}(n^2)$ time. Note that (2) has demand weights for every pair of vertices, so $\Omega(n^2)$ space is required to even explicitly write it down. On the other hand, each $\mathbf{v}_x \in \mathbb{R}^{O(\log n)}$, so the weights $\|\mathbf{v}_x - \mathbf{v}_y\|^2$ are all implicitly stored in only $O(n \log n)$ space. Therefore, making use of the additional geometric structure of these instances is crucial to achieving sub-quadratic time. Implicit in all of [15, 5, 20] is a specialized algorithm to approximately solve (2). The actual algorithm used is the same in all three, and those algorithms differ only in their strategy for the embedding player.

In the next two subsections, we briefly sketch the single-commodity and multicommodity flow based algorithms of [5], and then describe how we tie the two together. In particular, our algorithm is essentially an "algorithmetization" of the multicommodity flow algorithm's analysis. For the rest of the section, suppose we have an embedding V with $\sum_{x<y} \|\mathbf{v}_x - \mathbf{v}_y\|^2 = n^2$, and let us further assume that the points are unique and $\|\mathbf{v}_x\| \leq 1$ for all x; i.e., the diameter is not much more than the average distance.

2.1 Using Single-Commodity Flows

Consider first the absolute simplest case, where $d = 1$ and the points are simply numbers in $[-1, 1]$. It is easy to see that since the points are in $[-1, 1]$, unique, and have average squared-distance $\Omega(1)$, there must be some interval $[a, b]$, where $b - a = \Omega(1) =: \sigma$ and the set of points to the left of a, $A = \{x : \mathbf{v}_x \leq a\}$ and to the right of b, $B = \{y : \mathbf{v}_y \geq b\}$ have $|A| = |B| = \Omega(n) =: 2cn$. A natural way to try to push flow far along this line would be to shrink A and B down to single vertices and then compute a max-flow from A to B.

`FlowAndCut`$(\kappa, c, w_1, \ldots, w_n \in \mathbb{R})$:

- Sort $\{w_x\}$, let A be the $2cn$ nodes x with least w_x and B be those with greatest w_y.
- Add two vertices s, t. Connect s to each $x \in A$ and t to each $y \in B$ with edges of capacity κ.
- Output the max-flow/min-cut for $s - t$.

Consider invoking `FlowAndCut`$(\kappa, c, \mathbf{v}_1, \ldots, \mathbf{v}_n)$ with $\kappa = c^{-1}\sigma^{-2}$. If the max-flow is at least κcn, then since all flow must cross the gap $[a, b]$, we have pushed κcn units of flow across a squared-distance of σ^2, achieving a solution D with $\Phi(V, D) \geq (\kappa cn\sigma^2)/n = 1$. Otherwise, if the min-cut is at most κcn, then at most cn of the added κ-capacity edges are cut, so at least cn vertices must remain on each side and the cut has expansion at most κ. That is, for dimension one a $\kappa = O(1)$ approximation is obtained.

The approach of [5, 20] is to reduce the general case to the one-dimensional case by picking a random standard normal vector $\mathbf{u}$ and projecting each $\mathbf{v}_x$ along $\mathbf{u}$, yielding the 1-dimensional embedding $w_x = \mathbf{v}_x \cdot \mathbf{u}$. The fact that the points are in the unit ball and have average distance $\Omega(1)$ implies that with probability $\Omega(1)$, there is a gap $[a, b]$ with $b - a = \Omega(1) = \sigma$ as before. Applying the previous analysis, we either find a cut of expansion $O(1)$ or a flow with $\sum_{x<y} D_{xy}(w_x - w_y)^2 \geq n$. Then, the Gaussian tail ensures that distances

could not have been stretched too much along $\mathbf{u}$: with high probability $(w_x - w_y)^2 \leq O(\log n)\|\mathbf{v}_x - \mathbf{v}_y\|^2$ for every pair x, y. Thus, $\Phi(V, D) \geq \Omega(1/\log(n))$, yielding an $O(\log n)$ approximation.

2.2 Using Multi-Commodity Flows

Arora, Rao, and Vazirani showed that, if a *best* response D^* to V has $\Phi(V, D^*) \leq 1$, one can find a cut of expansion $O(\sqrt{\log n})$. Supposing the optimal solution to (2) has value at most n, there must be a solution to the dual problem of value at most n. The dual assigns lengths $\{w_e\}$ to the edges of G, aiming to minimize $\sum_e G_e w_e$ subject to the constraints that the shortest-path distances between each x, y under $\{w_e\}$ are at least $\|\mathbf{v}_x - \mathbf{v}_y\|^2$. Arora and Kale show the existence of such a dual solution implies that projecting the points along a random $\mathbf{u}$ and running `FlowAndCut`with $\kappa = \Theta(\sqrt{\log n})$ must yield a cut of capacity at most κcn with probability $\Omega(1)$.

If not, then a flow of value at least κcn is returned for $\Omega(1)$ of the directions $\mathbf{u}$ along which A and B are σ-separated. For simplicity, assume that the flows actually correspond to a matching between A and B. That is, each x has either zero flow leaving, or else has exactly κ flow going to a unique y along a single path. On the one hand, that matching is routed in G along cn flowpaths, each carrying flow κ. On the other hand, the total volume of G is only $\sum_e G_e w_e = n$, so $\Omega(n)$ of those flowpaths must have length at most $O(1/\kappa)$ under $\{w_e\}$.

For each $\mathbf{u}$, let $M(\mathbf{u})$ be the matching consisting of those demand pairs routed along such short paths. Then, according to the following definition, M is an $(\Omega(1), \Omega(1))$-matching-cover.

Definition 2.1. *A (σ, δ)-**matching-cover** for an embedding $\{\mathbf{v}_x\}$ is a collection $\{M(\mathbf{u})\}_{\mathbf{u}\in\mathbb{R}^d}$ of directed matchings satisfying the following conditions.*

- *Stretch: $(\mathbf{v}_y - \mathbf{v}_x) \cdot \mathbf{u} \geq \sigma$ for all $(x, y) \in M(\mathbf{u})$*
- *Skew-symmetry: $(x, y) \in M(\mathbf{u})$ iff $(y, x) \in M(-\mathbf{u})$*
- *Largeness: $\mathbf{E}_{\mathbf{u}}[|M(\mathbf{u})|] \geq \delta n$*

For a list of vectors $\mathbf{u}_1, \ldots, \mathbf{u}_R$, let $M(\mathbf{u}_1, \ldots, \mathbf{u}_R)$ denote the graph that contains edge (x, y) iff there exist $x_0, \ldots, x_R$ with $x_0 = x, x_R = y$ and $(x_{r-1}, x_r) \in M(\mathbf{u}_r)$ for all $r \leq R$. For the empty list, let $M()$ denote the graph where each vertex has a directed self-loop. Note that $M(\mathbf{u}_1, \ldots, \mathbf{u}_R)$ is not a matching, but rather a graph with maximum in-degree and out-degree one.

Furthermore, M has the property that for each edge $(x, y) \in M(\mathbf{u})$, the distance between x and y under $\{w_e\}$ is at most $O(1/\kappa)$. The following theorem holds for M.

Theorem 2.2 ([16], refining [7])**.** *Let M be a $(\Omega(1), \Omega(1))$-matching-cover for $\{\mathbf{v}_x\}$. Then, there are vertices x, y and $\mathbf{u}_1, \ldots, \mathbf{u}_R$ where $R \leq O(\sqrt{\log n})$ such that $(x, y) \in M(\mathbf{u}_1, \ldots, \mathbf{u}_R)$ and $\|\mathbf{v}_x - \mathbf{v}_y\|^2 \geq L$.*

In other words, there are vertices x, y with $\|\mathbf{v}_x - \mathbf{v}_y\|^2 \geq L$ that are only R matching hops away in M. Applying theorem 2.2, there are vertices x, y with $\|\mathbf{v}_x - \mathbf{v}_y\|^2 \geq L$ but of distance only $RO(1/\kappa)$ under $\{w_e\}$. Choosing $\kappa = O(R/L) = O(\sqrt{\log n})$ yields a contradiction to the assumption that $\{w_e\}$ is dual feasible.

2.3 Results.

Our improvement comes from being able to achieve an $O(\sqrt{\log n})$ gap between cut and flow solutions, as in the latter case, while still only using single-commodity flows, as in the former case. Recall the case of $d > 1$ was reduced to the $d = 1$ case by projecting along a random vector and bounding the squared-stretch by $O(\log n)$. Indeed, the stretch could be nearly that much, so simply pushing flow along a single direction will not allow us to achieve anything better; in fact, that is the main idea behind our lower bound for the cut-matching game.

To do better, we need to do something more sophisticated than simply push flow along a single direction. A natural idea is to try to pick several directions $\mathbf{u}_1, \ldots, \mathbf{u}_R$, push flow along each of them, and then try to glue the flows together to actually push flow far away globally. One motivation for such an approach is that it seems to be the next simplest thing to do, following that of using only a single direction. The second and most crucial motivation is to observe that *such an approach is strongly suggested by the analysis for the multicommodity flow algorithm just sketched.* To see that, suppose the typical flowpath along a random $\mathbf{u}$ routes between points of squared-distance Δ. Theorem 2.2 says we can always augment $R = O(\sqrt{\log n})$

such flowpaths to route demand between points of squared-distance $L = \Omega(1)$, at the cost of possibly raising congestion by a factor of R. Thus, either $\Delta \geq L/R = \Omega(1/\sqrt{\log n})$, or else augmenting together R typical flowpaths and scaling down by R maintains feasibility and increases the objective of (2).

Unfortunately, theorem 2.2 doesn't say anything at all about *finding* such directions $\mathbf{u}$, or whether the same $\mathbf{u}_1, \ldots, \mathbf{u}_R$ will simultaneously work for many vertices. To analyze such an algorithm, we need a stronger, algorithmic version of theorem 2.2. Our main technical contribution is such a theorem.

Theorem 2.3. *For any $1 \leq R \leq \Theta(\sqrt{\log(n)})$, there is $L \geq \Theta(R^2/\log(n))$ and an (efficiently sample-able) distribution $\mathcal{D}$ over $(\mathbb{R}^d)^{\leq R}$ with the following property.*

If M is an $(\Omega(1), \Omega(1))$-matching-cover for $\{\mathbf{v}_x\}$, then the expected number of edges (x, y) with $(x, y) \in M(\mathcal{D})$ and $\|\mathbf{v}_x - \mathbf{v}_y\|^2 \geq L$ is at least $e^{-O(R^2)}n$.

Using theorem 2.3 and choosing $R = \Theta(\sqrt{\varepsilon \log(n)})$, $L = \Theta(\varepsilon)$, we simply sample $\mathbf{u}_1, \ldots, \mathbf{u}_R$ from $\mathcal{D}$, and let our final flowpaths be the concatenation of those along $\mathbf{u}_1, \ldots, \mathbf{u}_R$. On average, we get $n^{1-\varepsilon}$ such paths, and thus have *simultaneously* routed $n^{1-\varepsilon}$ paths between points of squared distance L using only R single-commodity flows. Using an iterative re-weighing scheme and repeating $O(n^{\varepsilon} \log^{O(1)} n)$ times, we achieve a feasible flow and an approximation ratio of $O(R/L) = O(\sqrt{\log(n)/\varepsilon})$.

That is, to push flow far away, we sample $\mathbf{u}_1, \ldots, \mathbf{u}_R$ from $\mathcal{D}$ and then iteratively push flow along each direction. The distribution $\mathcal{D}$ will essentially consist of picking a random direction $\mathbf{u}_1$, and then choosing $\mathbf{u}_{r+1}$ to be a $1 - 1/R$-correlated copy of $\mathbf{u}_r$; i.e., a vector extremely close to $\mathbf{u}_r$. Because $\mathbf{u}_{r+1}$ and $\mathbf{u}_r$ are so close, it is intuitively clear and easy to argue that *if* flow gets pushed along at each step, it must be pushed far away, as the projections along each $\mathbf{u}_r$ will essentially add together. The somewhat counterintuitive fact is that flow actually does get pushed further along in this manner. Even though $\mathbf{u}_r$ and $\mathbf{u}_{r+1}$ are extremely close together, a significant fraction of vertices that were in the "sink set" along $\mathbf{u}_r$ will be in the "source set" along $\mathbf{u}_{r+1}$. That phenomenon is a consequence of measure concentration.

3 The Algorithm

While we found it most convenient to discuss expander flows and the corresponding game in the context of the SPARSEST CUT problem, our algorithm applies most directly to BALANCED SEPARATOR, which has a similar SDP relaxation and game. Roughly, the difference is that in the BALANCED SEPARATOR case the embedding player must choose an embedding for which the maximum squared distance between points is not much larger than the average. When the average distance is $\Theta(1)$, this is equivalent to the requirement that $\|\mathbf{v}_x\| \leq O(1)$ assumed earlier in section 2. The reduction from SPARSEST CUT to BALANCED SEPARATOR is well-known, and in fact, the unbalanced case is "easy" in the sense that if the cut found is unbalanced, it will be an $O(1)$ approximation to the sparsest cut[7]. In particular, Arora and Kale show that one can either obtain an $O(1)$ cut/flow gap with a single max-flow, or else reduce the problem to the balanced case by finding $\Omega(n)$ points in a ball of radius $O(1)$ that are still spread-out within that ball; for details, we refer the reader to [5].

The precise statement of the results sketched in section 2 is the following main lemma of [5].

Lemma 3.1 ([5])**.** *Let $U \subseteq [n]$ be a set of nodes. Suppose we are given vectors $V = \{\mathbf{v}_x\}_{x \in U}$ of length at most $O(1)$ such that $\sum_{x,y \in U} \|\mathbf{v}_x - \mathbf{v}_y\|^2 = n^2$.*

- *There is an algorithm that uses $O(1)$ expected max-flow computations and outputs either a demand graph D on U of max-degree $O(\log(n))$ that is routable in G with $\Phi(V, D) \geq 1$ or a balanced cut of expansion $O(\log n)$.*
- *There is an algorithm that uses a single multicommodity flow computation and $O(1)$ expected max-flow computations and outputs either a demand graph D on U of max-degree $O(1)$ that is routable in G with $\Phi(V, D) \geq 1$ or a balanced cut of expansion $O(\sqrt{\log n})$.*

The importance of the degree is for the running time; if each D^t has max-degree β, then the total number of iterations needed is $O(\beta \log(n))$[5]. To prove theorem 1.1, we replace lemma 3.1 with the following.

Lemma 3.2. *Let U, V be as in lemma 3.1. For any $\varepsilon \in [O(1/\log(n)), \Omega(1)]$, there is an algorithm that uses $O(n^{\varepsilon} \log^{O(1)}(n))$ expected max-flow computations and outputs either a demand graph D on U of max-degree $O(1/\varepsilon)$ routable in G with $\Phi(V, D) \geq 1$ or a balanced cut of expansion $O(\sqrt{\log(n)/\varepsilon})$.*

For the rest of this section, we prove lemma 3.2. We first immediately try to find a cut, using `FlowAndCut`. The parameters c, σ are set by the following lemma.

Lemma 3.3 ([7]). *Let U, V be as in lemma 3.1. Then, there exist $c, \sigma, \gamma = \Omega(1)$ so for a random $\mathbf{u}$, with probability at least γ the sets A, B in $\texttt{FlowAndCut}(\cdot, c, \{\mathbf{v}_x \cdot \mathbf{u}\}_{x \in U})$ have $(\mathbf{v}_y - \mathbf{v}_x) \cdot \mathbf{u} \geq \sigma$ for all $x \in A, y \in B$.*

Let us call the $\mathbf{u}$ described by lemma 3.3 *good*, and set $\delta = \gamma c/16$. Let $\varepsilon \in [O(1/\log(n)), \Omega(1)]$ be given so that $R = O(\sqrt{\varepsilon \log n})$ yields an expected size bound of $n^{1-\varepsilon}$ in theorem 2.3. Set $L = \Omega(\varepsilon)$ as in theorem 2.3, $\kappa = 24R/cL$, and $\beta = 12/cL$. The following easy lemma was sketched in section 2.

Lemma 3.4 ([15, 5]). *If $\texttt{FlowAndCut}(\kappa, c, \ldots)$ returns a cut of capacity at most κcn, then the cut is cn-balanced and has expansion at most κ.*

We sample $O(\log(n))$ independent $\mathbf{u}$, and run $\texttt{FlowAndCut}(\kappa, c, \{\mathbf{v}_x \cdot \mathbf{u}\})$. If we ever find a cut of capacity at most κcn, we immediately output it and stop, yielding a balanced cut of expansion $\kappa = O(\sqrt{\log(n)/\varepsilon})$. Otherwise, with very high probability, we are in the situation where there are at least $\gamma/2$ good $\mathbf{u}$ for which a flow of value at least κcn is returned. In the latter scenario, we will find a flow D with $\Phi(V, D) \geq 1$.

3.1 Finding a Flow

We efficiently find a solution to the maximum multicommodity flow problem

$$\begin{aligned} &\max \sum_{x<y} D_{xy} \|\mathbf{v}_x - \mathbf{v}_y\|^2 \\ &s.t. \quad F \leq G, \quad \max_x \deg_D(x) \leq \beta \end{aligned} \tag{3}$$

of value at least n. The dual assigns lengths $\{w_e\}$ to edges and $\{w_x\}$ to the vertices, with the constraint that the shortest path distance from x to y under these lengths dominate $\|\mathbf{v}_x - \mathbf{v}_y\|^2$.

$$\min \sum_e G_e w_e + \sum_x \beta w_x$$

$$s.t. \quad \forall p : x \leftrightarrow y \quad w_x + w_y + \sum_{e \in p} w_e \geq \|\mathbf{v}_x - \mathbf{v}_y\|^2$$

We use the *multiplicative weights* framework to approximately solve (3).

Theorem 3.5 ([21, 13, 4]). *Let $A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$ with $b > 0$, and consider the following iterative procedure to find an approximate solution to $Ax \leq b$.*

Initialize $y^1 \in \mathbb{R}^m$ to the all-1s vector. On iteration t, query an oracle that returns x^t such that $0 \leq Ax^t \leq \rho b$ and $y^t \cdot Ax^t \leq y^t \cdot b$, and then update

$$y_j^{t+1} \leftarrow \left(1 + \eta \frac{(Ax^t)_j}{\rho b_j}\right) y_j^t$$

If $0 < \eta < 1/2$, then after $T = \rho \eta^{-2} \log(n)$ iterations, $A\left(\frac{x^1 + \cdots + x^T}{T}\right) \leq (1 + 4\eta)b$.

We use theorem 3.5 with $\eta = 1/4$, initializing the dual variables $\{w_e\}, \{w_x\}$ and updating them accordingly. On iteration t, we find a flow (F^t, D^t) of objective value $2n$ that violates the constraints by at most a factor of $\rho = O(n^{2\varepsilon} \log n)$ and

$$\sum_e w_e F_e^t + \sum_x w_x \deg_{D^t}(x) \leq \sum_e w_e G_e + \sum_x w_x \beta \tag{4}$$

After $T = O(n^{2\varepsilon} \log^2(n))$ rounds, scaling the average flow down by 2 yields a feasible flow of objective value n. Noting that (4) and the algorithm of theorem 3.5 are invariant to scaling of the dual variables, for convenience we will also scale them on each iteration so that $\sum_e w_e G_e + \sum_x w_x \beta = 2n$. In that case, any flow of objective value $2n$ that only routes along violated or tight paths (those $p : x \leftrightarrow y$ for which $\sum_{e \in p} w_e + w_x + w_x \leq \|\mathbf{v}_x - \mathbf{v}_y\|^2$) satisfies (4). In our algorithm, we will only route flow along paths $p : x \leftrightarrow y$ for which $\|\mathbf{v}_x - \mathbf{v}_y\|^2 \geq L$, and $w_x, w_y, \sum_{e \in p} w_e \leq L/3$.

All flows will come from augmenting flows returned by `FlowAndCut`, where we identify single-commodity flows in $G \cup \{s, t\}$ with multicommodity flows in G in the obvious way. If F is an acyclic $s - t$ flow in $G \cup \{s, t\}$, it is well-known that F can be decomposed into at most m flowpaths. While computing such a decomposition could require $\Omega(nm)$ time, fortunately we need only pseudo-decompose flows in the following sense.

Definition 3.6. *If F is an acyclic $s - t$ flow in $G \cup \{s, t\}$ with a flow decomposition $((f_i, p_i))_{i \leq m}$, then a list $P = ((f_i, s_i, t_i, \ell_i))_{i \leq m}$ where $p_i = s, s_i, \dots, t_i, t$ and $\sum_{e \in p_i} w_e = \ell_i$ is a **pseudo-decomposition** of F. That is, a pseudo-decomposition is a list containing the amount of flow, second vertex, second-to-last vertex, and length of each flowpath.*

The following two lemmas are easy applications of dynamic trees(see [22]).

Lemma 3.7. *Given a flow F on $G \cup \{s, t\}$, a pseudo-decomposition can be computed in $O(m \log n)$ time.*

Lemma 3.8. *Given a flow F on $G \cup \{s, t\}$, and a desired scaling vector $(\alpha_1, \dots, \alpha_m)$, we can compute the flow F' with decomposition $\{(\alpha_k f_k, p_k)\}$ in $O(m \log n)$ time.*

Lemma 3.8 allows us to efficiently cherry-pick "good" flowpaths from the flows returned by `FlowAndCut`.

In their analysis, Arora and Kale round the flows returned by `FlowAndCut` to matchings. We do the same, with a small change to ensure doing so does not raise congestion by too much.

`Matching`($\mathbf{u}$)

- Call `FlowAndCut`($\kappa, c, \{\mathbf{v}_x \cdot \mathbf{u}\}$) and pseudo-decompose the resulting flow into P. Set $M = \emptyset$.
- Throw away any $(f_i, s_i, t_i, \ell_i) \in P$ with $(\mathbf{v}_{t_i} - \mathbf{v}_{s_i}) \cdot \mathbf{u} < \sigma$, $f_i < \kappa cn/4m$, $w_{s_i} > L/3$, $w_{t_i} > L/3$, or $\ell_i > L/3R$.
- Greedily match the remaining pairs: iteratively pick $(f_i, s_i, t_i, \ell_i) \in P$, add (s_i, t_i) to M, and remove any $(f_j, s_j, t_j, \ell_j) \in$ with $\{s_i, t_i\} \cap \{s_j, t_j\} \neq \emptyset$.
- Output M.

The following lemma is essentially the same as one used in [5], and follows by the choice of parameters. The congestion bound, which was not needed for their analysis but is needed for our algorithm, comes from the fact that `Matching` discards any flowpath with $f_i \leq \kappa cn/4m$ before scaling any remaining flows to 1.

Lemma 3.9. *`Matching` is a (σ, δ)-matching-cover. Furthermore, for each $\mathbf{u}$, the (unit-weighted) demands `Matching`($\mathbf{u}$) are simultaneously routable in G with congestion at most $4m/\kappa cn$ along flowpaths of length at most $L/3R$ under $\{w_e\}$.*

Proof. The symmetry and stretched properties hold by construction, so we need only establish the largeness property. Let $\mathbf{u}$ be a good direction for which the returned flow has value at least κcn, and let D be the corresponding demands. Since $\mathbf{u}$ is good, every demand pair is σ-separated along $\mathbf{u}$. Each $x \in U$ has degree at most κ and the total degree is at least $2\kappa cn$. Deleting each path with $f_i \leq \kappa cn/4m$ removes at most $\kappa cn/4$ total flow. Since $\sum_x w_x \beta \leq 2n$ and $\beta = 12/cL$, at most $cn/4$ vertices can have $w_x > L/3$; deleting them removes at most $\kappa cn/4$ units of flow. Finally, since the original flow was feasible in the original graph,

$$\sum_p f_p \ell_p = \sum_e w_e \sum_{p \ni e} f_p \leq \sum_e w_e G_e \leq 2n$$

Since $\kappa = 24R/cL$, at most $\kappa cn/4$ units can flow along paths longer than $L/3R$.

In total, the second step of `Matching` removes at most $3\kappa cn/4$ units of flow, so at least $\kappa cn/2$ total degree survives. Each greedy matching step decreases the total degree by at most 4κ, so at least $cn/8$ pairs must get matched. Thus, the expected size of `Matching`($\mathbf{u}$) is at least $(\gamma/2)(cn/8) = \delta$.

For the congestion bound, we threw away all paths with flow less than $\kappa cn/4m$, so scaling the remaining paths to 1 yields a flow with congestion at most $4m/\kappa cn$. □

On each iteration, we sample $\mathbf{u}_1, \dots, \mathbf{u}_R$ from the distribution $\mathcal{D}$ of theorem 2.3 and call `Matching`($\mathbf{u}_r$). Let D' be the unit-weighted graph with an edge (x, y) for each $(x, y) \in$ `Matching`($\mathbf{u}_1, \dots, \mathbf{u}_R$) with $\|\mathbf{v}_x - \mathbf{v}_y\|^2 \geq L$. By theorem 2.3, the expected size of D' is at least $n^{1-\varepsilon}$, so after $n^\varepsilon/2$ expected trials, we have $|D'| \geq n^{1-\varepsilon}/2$. Applying lemma 3.8 again R times, we can compute a flow F' that routes D' in G with congestion $R(4m/\kappa cn) = O(\log(n)/L)$, since $m = O(n \log n)$ by assumption. Note also that D' has max-degree 2.

Then, D' achieves an objective value of at least $|D'|L$, so scaling up by $2n/|D'|L$ yields a solution of value $2n$ that satisfies (4) and congests edges by at most an $O(n^\varepsilon \log n)$ factor. Since $\beta = 12/cL = \Omega(1/\varepsilon)$, the degree constraints are also violated by at most an $O(n^\varepsilon)$ factor. The running time is dominated by flow computations, of which there are an expected $O(Rn^\varepsilon)$ in each of $O(n^\varepsilon \log^2(n))$ iterations, for a total of $O(n^{2\varepsilon} \log^{5/2}(n))$ expected max-flows.

4 Proof of Theorem 2.3

Let M be a (σ, δ)-matching cover. We identify M with a weighted directed graph, where edge (x, y) is has weight $\mathbf{Pr}_{\mathbf{u}}[(x, y) \in M(\mathbf{u})]$. The skew-symmetry condition ensures the weights of (x, y) and (y, x) are the same, as are the in-degree and out-degree of each x. The total out-degree of M is at least δn by assumption. Following [7], we first prune M to a more uniform version by iteratively removing any vertex of out-degree less than $\delta/4$. Doing so preserves skew-symmetry, and at least $\delta n/2$ out-degree remains. It follows that we are left with a matching cover on vertices X, with $|X| \geq \delta n/2$ and every $x \in X$ has out-degree at least $\delta/4$. The pruned M is a $(\sigma, \delta/4)$-uniform-matching-cover.

Definition 4.1. *A (σ, δ)-**uniform-matching-cover** of $X \subseteq [n]$ is a $(\sigma, 0)$-matching-cover where every $x \in X$ has in-degree at least δ in M.*

4.1 Chaining and Measure Concentration

Let $y \in X$, and let A be the set of $\mathbf{u}$ for which y has an out-edge in $M(\mathbf{u})$. The main idea behind the proof of theorem 2.2 is the following. Since A and $-A$ are two sets of measure $\Omega(1)$, the isoperimetric profile of Gaussian space implies there must be many $\mathbf{u} \in A, \hat{\mathbf{u}} \in -A$ that are very close: $\|\mathbf{u} - \hat{\mathbf{u}}\| \leq O(1)$ (we remark that [7] uses the uniform measure on the sphere, but the same analysis holds for Gaussians after scaling various quantities by $\sqrt{d}$). Choose x, z with $(x, y) \in M(\hat{\mathbf{u}})$, $(y, z) \in M(\mathbf{u})$ and observe that

$$\begin{aligned}(\mathbf{v}_y - \mathbf{v}_x) \cdot \mathbf{u} &= (\mathbf{v}_y - \mathbf{v}_x) \cdot \hat{\mathbf{u}} - (\mathbf{v}_y - \mathbf{v}_x) \cdot (\hat{\mathbf{u}} - \mathbf{u}) \\ &\geq \sigma - \|\mathbf{v}_x - \mathbf{v}_y\| \|\hat{\mathbf{u}} - \mathbf{u}\|\end{aligned}$$

Thus, either $\|\mathbf{v}_x - \mathbf{v}_y\| \geq \Omega(\sigma)$, or else $(\mathbf{v}_y - \mathbf{v}_x) \cdot \mathbf{u} \geq \sigma/2$. In the former case, a matching edge joins two points of distance $\Omega(1)$. In the latter case, replacing the edge $(y, z) \in M(\mathbf{u})$ with (x, z) yields an edge with $(\mathbf{v}_z - \mathbf{v}_x) \cdot \mathbf{u} \geq (3/2)\sigma$. By an inductive argument, the chaining case can be repeated until an edge connects two points of distance $\Omega(1)$. On the one hand, after R chaining steps, we have pairs of points that are R matching-hops apart, $O(1)$ distance apart, and have projection $\Theta(R)$. On the other hand, with high probability, no pair of distance $\Theta(1)$ has projection $\Theta(\sqrt{\log n})$, so the process must end after $\Theta(\sqrt{\log n})$ steps.

To turn the argument into an algorithm, we choose a sequence of highly correlated directions $\mathbf{u}_1, \ldots, \mathbf{u}_R$. For $R \geq 1$ and $0 \leq \rho \leq 1$, let $\mathcal{N}_\rho^R$ be the distribution of $\mathbf{u}_1, \ldots, \mathbf{u}_R$ defined by choosing a standard normal $\mathbf{u}_1$, and then choosing each $\mathbf{u}_{r+1} \sim_\rho \mathbf{u}_r$ to be a ρ-correlated copy of $\mathbf{u}_r$. That is, each of the d coordinate vectors $(\mathbf{u}_{1,i}, \ldots, \mathbf{u}_{R,i})$ are independently distributed as multivariate normals with covariance matrix $\Sigma_{r,r'} = \rho^{|r-r'|}$. In fact, simply setting $\mathcal{D} = \mathcal{N}_{1-1/R}^R$ achieves theorem 2.3 for $R \leq O(\log^{1/3}(n))$ and size bound of $e^{-O(R^3)}n$. The barrier is essentially the same as the one that limited the original analysis of [7] to $R = O(\log(n)^{1/3})$. To overcome that barrier, we algorithmetize Lee's improvement[16] by independently sampling uncorrelated $\mathbf{w}_1, \ldots, \mathbf{w}_R$, and then shuffling the two lists together. The idea is that the highly correlated $\mathbf{u}_r$ will give us long stretch, while the $\mathbf{w}_r$ will greatly increase the probability of forming a long chain, at the cost of losing some stretch. The sampling algorithm is:

`Sample`(R, ρ)

- Pick $\mathbf{u}_1, \ldots, \mathbf{u}_R \sim \mathcal{N}_\rho^R$, $\mathbf{w}_1, \ldots, \mathbf{w}_R \sim \mathcal{N}_0^R$.
- Pick a random shuffling of the two lists, pick a random $r \leq R$, and output the first r elements of the shuffled list.

The reason for the randomness is to keep the algorithm trivial, leaving the work to our analysis. We show that there exists a particular shuffling and $r \leq R$ for which `Sample` is good; by randomly guessing, we lose at most a 2^{R+1} factor in our final expectation bound, which is negligible relative to the $e^{-O(R^2)}n$ bound we are aiming for.

Our proof of theorem 2.3 closely follows Lee's proof of theorem 2.2, the main difference being the use of a stronger isoperimetric inequality. The the standard isoperimetric inequality says that if A is a set of large measure, then for almost points $\mathbf{u}$, a small ball around $\mathbf{u}$ has non-empty intersection with A. We use a stronger version, saying that if A is a set of large measure, then for almost all points $\mathbf{u}$, a small ball around $\mathbf{u}$ has a significantly large intersection with A.

Lemma 4.2. *Let $A \subseteq \mathbb{R}^d$ have Gaussian measure $\delta > 0$. If $\mathbf{u}, \hat{\mathbf{u}}$ are ρ-correlated with $0 \leq \rho < 1$, then*

$$\mathbf{Pr}_{\mathbf{u}}\left[\mathbf{Pr}_{\hat{\mathbf{u}}}\left[\hat{\mathbf{u}} \in A\right] < (\varepsilon\delta)^{1/(1-\rho)}\right] < \varepsilon$$

Lemma 4.2 is an easy corollary of Borell's reverse hypercontractive inequality [9]; we include a short proof in appendix A. Applications of Borell's result to strong isoperimetric inequalities appear in [18], and we follow the proofs of similar lemmas there.

4.2 Definitions

For a matching-cover M and a distribution $\mathcal{D}$ over $\mathbb{R}^*$, let $M(\mathcal{D})$ be the random graph $M(\mathbf{u}_1, \ldots, \mathbf{u}_r)$ where $\mathbf{u}_1, \ldots, \mathbf{u}_r \sim \mathcal{D}$. For a random graph $\mathcal{G}$ and sets $S, T \subseteq [n]$, let $\mu_{\mathcal{G}}(S, T)$ be the expected number of edges from S to T in $\mathcal{G}$. We say S is γ-*connected* to T in $\mathcal{G}$ if $\mu_{\mathcal{G}}(S, T) \geq \gamma$. For singleton sets, we omit braces and write $\mu_{\mathcal{G}}(x, y)$ for the probability that the edge (x, y) is in $\mathcal{G}$.

Two sets that will be useful are,

$$\mathsf{Ball}[x; \ell] = \{y : \|\mathbf{v}_x - \mathbf{v}_y\| \leq \ell\}$$
$$\mathsf{Stretch}[x, \sigma, \mathbf{u}] = \{y : (\mathbf{v}_y - \mathbf{v}_x) \cdot \mathbf{u} \geq \sigma\}$$

We will also work with collections of distributions $\mathcal{D} = \{\mathcal{D}(\mathbf{u})\}$ over $\mathbb{R}^*$ parameterized by $\mathbf{u}$. Such a collection is itself associated with the distribution induced by sampling a standard normal $\mathbf{u}$ and then sampling from $\mathcal{D}(\mathbf{u})$.

Definition 4.3. *Let $\mathcal{D}$ be a distribution collection. We say a vertex x is $(\sigma, \delta, \gamma, \ell)$-**covered** in $M(\mathcal{D})$ if for least δ of $\mathbf{u}$, x is γ-connected to $\mathsf{Stretch}[x, \sigma, \mathbf{u}] \cap \mathsf{Ball}[x; \ell]$ in $M(\mathcal{D}(\mathbf{u}))$.*

4.3 Cover Lemmas

Our goal is to exhibit a distribution $\mathcal{D}$ such that many vertices x are well-connected to $X \setminus \mathsf{Ball}[x; \sqrt{L}]$ in $M(\mathcal{D})$. To do so, we inductively construct particular distribution collections $\mathcal{D}^r$ such that many vertices x are either $e^{-O(Rr)}$-connected to $X \setminus \mathsf{Ball}[x; \sqrt{L}]$ in $M(\mathcal{D}^r)$, or else are $(\Omega(r), \Omega(1), e^{-O(rR)}, \sqrt{L})$-covered by $M(\mathcal{D}^r)$.

We begin with a trivial bound on how much points can be covered.

Lemma 4.4. *For $\ell, \gamma, \delta > 0$ and arbitrary $M, \mathcal{D}$, no vertex is $(\ell\sqrt{2\log(n/\delta)}, \delta, \gamma, \ell)$-covered by $M(\mathcal{D})$.*

Proof. For any $y \in \mathsf{Ball}[x; \ell]$, the probability that $(\mathbf{v}_y - \mathbf{v}_x) \cdot \mathbf{u} \geq \beta$ is at most $\exp(-\beta^2/2\ell^2) \leq \delta/n$ for $\beta = \ell\sqrt{2\log(n/\delta)}$. It follows that the probability that $\mathsf{Stretch}[x, \ell\sqrt{2\log(n/\delta)}, \mathbf{u}] \cap \mathsf{Ball}[x; \ell]$ is non-empty is at most $(n-1)\delta/n < \delta$. □

The next lemma says that if a vertex x is connected by $\mathcal{D}'$ to a set S of vertices that are covered by $\mathcal{D}$, then x is covered by the concatenation of $\mathcal{D}'$ and $\mathcal{D}$.

Lemma 4.5. *Let S be a set of vertices such that each $y \in S$ is $(\sigma, \delta, \gamma, \ell)$-covered by $M(\mathcal{D})$. Let x be a vertex with $\mu_{M(\mathcal{D}')}(x, S \cap \mathsf{Ball}[x; \ell']) \geq \gamma'$. Then, x is $(\sigma - \sqrt{2\ell' \log(2/\delta)}, \delta/4, \gamma\gamma'\delta/4, \ell + \ell')$-covered by $\mathcal{D}''(\mathbf{u}) = \mathcal{D}', \mathcal{D}(\mathbf{u})$.*

Proof. Let Γ be the distribution of x's out-neighbor in $M(\mathcal{D}')$, conditioned on $S \cap \mathsf{Ball}[x;\ell']$. For each $y \in S$, let A_y be the set of $\mathbf{u}$ for which y is γ-connected to $\mathsf{Stretch}[y,\sigma,\mathbf{u}] \cap \mathsf{Ball}[y;\ell]$ in $M(\mathcal{D}(\mathbf{u}))$.

For any fixed $y \in \Gamma$, the quantity $(\mathbf{v}_x - \mathbf{v}_y)\cdot\mathbf{u}$ is normal with mean zero and variance $\|\mathbf{v}_y - \mathbf{v}_x\|^2 \le \ell'^2$, so the probability (over $\mathbf{u}$) that $y \in \mathsf{Stretch}[x,-\beta,\mathbf{u}]$ is at least $1-\exp(-\beta^2/2\ell'^2) \ge 1-\delta/2$ for $\beta = \sqrt{2\ell'\log(2/\delta)}$. Then, for at least $\delta/2$ of $\mathbf{u}$, we have $y \in \mathsf{Stretch}[x,-\beta,\mathbf{u}]$ and $\mathbf{u} \in A_y$. By averaging, for at least $\delta/4$ of $\mathbf{u}$, at least $\delta/4$ of $y \sim \Gamma$ have $y \in \mathsf{Stretch}[x,-\beta,\mathbf{u}]$ and $\mathbf{u} \in A_y$. It follows that x is $(\sigma-\beta,\delta/4,\gamma\gamma'\delta/4,\ell+\ell')$-covered by $\mathcal{D}''$. ☐

Our next lemma is the main chaining step.

Lemma 4.6. *Let M be a $(\sigma_0,\cdot)$-matching-cover, T be a set of vertices that are $(\sigma, 1-\delta/2,\gamma,\infty)$-covered in $M(\mathcal{D})$, and S a set of vertices that is $\delta|T|$-connected to T in M. Then, at least $\delta|T|/2$ vertices $x \in S$ are $(\sigma+\sigma_0, \delta|T|/4|S|, \gamma, \infty)$-covered by $\mathcal{D}'(\mathbf{u}) = \mathbf{u}, \mathcal{D}(\mathbf{u})$.*

Proof. Let M' be the subgraph of M consisting only of edges from S to T; by assumption the total degree in M' is at least $\delta|T|$. Further remove any edge $(x,y) \in M'(\mathbf{u})$ where $\mu_{M(\mathcal{D}(\mathbf{u}))}(y, \mathsf{Stretch}[y,\sigma,\mathbf{u}]) < \gamma$. The total in-degree remaining is at least $\delta|T|/2$, so there is a set $S' \subseteq S$ of at least $\delta|T|/2$ vertices that have out-degree at least $\delta|T|/4|S|$. Finally, note that if $(x,y) \in M'(\mathbf{u})$, then $\mathsf{Stretch}[y,\sigma,\mathbf{u},\infty] \subseteq \mathsf{Stretch}[x,\sigma+\sigma_0,\mathbf{u},\infty]$, so each $x \in S'$ is $(\sigma+\sigma_0,\delta|T|/4|S|,\gamma,\infty)$-covered by $\mathcal{D}'$. ☐

To apply lemma 4.6, we need to establish covers with δ very close to 1. Consider taking a collection $\mathcal{D}$ and then *smoothing* it by replacing $\mathcal{D}(\mathbf{u})$ with the average of $\mathcal{D}(\hat{\mathbf{u}})$ for nearby $\hat{\mathbf{u}}$. The next lemma shows that doing so boosts δ to nearly 1, in exchange for a loss in σ and γ.

Lemma 4.7. *Let x be $(\sigma,\delta,\gamma,\ell)$-covered by $\mathcal{D}$. Then, x is $(\rho\sigma - 4\ell\sqrt{\log(2/\delta)}, 1-2\delta, \delta^{2/(1-\rho)}\gamma/4, \ell)$-covered by $\mathcal{D}'(\mathbf{u}) = \mathcal{D}(\hat{\mathbf{u}})$ where $\hat{\mathbf{u}} \sim_\rho \mathbf{u}$.*

Proof. Let A be the set of $\hat{\mathbf{u}}$ for which x is γ-connected to $\mathsf{Stretch}[x,\sigma,\hat{\mathbf{u}}] \cap \mathsf{Ball}[x;\ell]$ in $M(\mathcal{D}(\hat{\mathbf{u}}))$. For each $\hat{\mathbf{u}}$, let $\Gamma(\hat{\mathbf{u}})$ be the distribution of x's out-neighbor in $M(\mathcal{D}(\hat{\mathbf{u}}))$, conditioned on $\mathsf{Stretch}[x,\sigma,\hat{\mathbf{u}}] \cap \mathsf{Ball}[x;\ell]$.

For any $\hat{\mathbf{u}}$ and $y \in \Gamma(\hat{\mathbf{u}})$, the quantity $(\mathbf{v}_y - \mathbf{v}_x)\cdot\mathbf{u}$ is normal with mean $\rho(\mathbf{v}_y - \mathbf{v}_x)\cdot\hat{\mathbf{u}} \ge \rho\sigma$ and variance $(1-\rho^2)\|\mathbf{v}_y-\mathbf{v}_x\|^2 \le 2(1-\rho)\ell^2$; it follows that $y \in \mathsf{Stretch}[x,\rho\sigma-\beta,\mathbf{u}]$ with probability at least $1-\exp(-\beta^2/4(1-\rho)\ell^2) \ge 1-(\delta/2)^{4/(1-\rho)}$ for $\beta = 4\ell\sqrt{\log(2/\delta)}$ over $\mathbf{u}$. By averaging, for at least $1-\delta$ $\mathbf{u}$, for at least $1-(2/\delta)(\delta/2)^{4/(1-\rho)}$ $\hat{\mathbf{u}} \sim_\rho \mathbf{u}$, we have $\mathbf{Pr}\left[\Gamma(\hat{\mathbf{u}}) \in \mathsf{Stretch}[x,\rho\sigma-\beta,\mathbf{u}]\right] \ge 1/2$. Call such pairs $(\mathbf{u},\hat{\mathbf{u}})$ good.

Applying lemma 4.2 to A, for at least $1-\delta$ $\mathbf{u}$, we have $\mathbf{Pr}[\hat{\mathbf{u}} \in A] \ge \delta^{2/(1-\rho)}$. All together, for at least $1-2\delta$ $\mathbf{u}$, with probability at least $\delta^{2/(1-\rho)} - (2/\delta)(\delta/2)^{4/(1-\rho)}$ we have both $\hat{\mathbf{u}} \in A$ and $(\mathbf{u},\hat{\mathbf{u}})$ good. In that case, $\mu_{M(\mathcal{D}(\hat{\mathbf{u}}))}(x, \mathsf{Stretch}[x,\rho\sigma-\beta,\mathbf{u}] \cap \mathsf{Ball}[x;\ell]) \ge \gamma/2$. The lemma follows by noting $(2/\delta)(\delta/2)^{4/(1-\rho)} \le \delta^{2/(1-\rho)}/2$. ☐

Combining the previous results, we prove the main inductive lemma.

Lemma 4.8. *Let M be a (σ,δ)-uniform-matching-cover of X where $\delta \le 1/4$. Let $\ell \le \sigma/2^7\sqrt{\log(1/\delta)}$ and $K \ge 1$. Then, one of the following must occur.*

1. *There are distribution collections $\mathcal{D}^0,\ldots,\mathcal{D}^K$ such that for every $k \le K$, at least $\delta^{6k}|X|$ vertices are $(k\sigma/4, \delta^8, \delta^{24Kk}, \ell)$-covered in $M(\mathcal{D}^k)$.*

2. *There is a distribution $\mathcal{D}^*$ such that at least $\delta^{6K}|X|$ vertices x are δ^{24K^2}-connected to $X \setminus B[x;\ell]$ in $M(\mathcal{D}^*)$. Furthermore, $\mathcal{D}^*$ is a shuffling of $\mathcal{N}^k_{1-1/K}$ with $\mathcal{N}^{k'}_0$ for some $k \le K$ and $k' \le 6K$.*

Proof. For $k=0$, every $x \in X$ is $(0,1,1,0)$-covered by $\mathcal{D}^0(\mathbf{u}) = ()$, the empty list.

Assuming case 1 holds for some $0 \le k < K$, let T_0 be those vertices that are $(k\sigma/4,\delta^8,\gamma,\ell)$-covered by $\mathcal{D}^k$, where $\gamma = \delta^{24Kk}$. We begin by finding a set S that is well-connected to T_0. Since at least $\delta|T_0|$ in-degree enters T_0 in M, by averaging either at least $\delta^{-1}|T_0|$ vertices have at least $\delta^2|T_0|/|X|$ out-degree into T_0 or else at least $\delta|T_0|$ vertices have at least δ^3 out-degree into T_0. In the former case, call that set T_1 and repeat, yielding sets $T_0,T_1,\ldots,T_t$ where each $y \in T_s$ has at least $\delta^2|T_{s-1}|/|X|$ out-degree into T_{s-1}. Let S be those vertices with out-degree at least δ^3 into T_t, so that $\delta|T_t| \le |S| \le \delta^{-1}|T_t|$. Let $\mathcal{D}' = \mathcal{N}^t_0$; by construction, each $y \in T_t$ has

$$\begin{aligned}\mu_{M(\mathcal{D}')}(y, T_0) &\geq \prod_{s=0}^{t-1} \delta^2 |T_s|/|X| \\ &\geq \delta^{2t - t(t-1)/2} |T_0|/|X| \\ &\geq \delta^{3+6k}\end{aligned}$$

Assuming case 2 does not hold by setting $\mathcal{D}^* = \mathcal{D}'$, there is a set $T \subseteq T_t$ of size at least $(1 - \delta^5)|T_t|$ such that each $y \in T$ has $\mu_{M(\mathcal{D}')}(y, T_0 \cap \mathsf{Ball}[y; \ell]) \geq \delta^{3+6k}/2 =: \gamma'$. It follows that at least $\delta^3|S| - \delta^5|T_t| \geq \delta^5|T_t|$ out-degree from S enters T.

Lemma 4.5 implies each $y \in T$ is $((k-1)\sigma/4, \delta^9, \gamma'', 2\ell)$-covered by $\mathcal{D}''(\mathbf{u}) = \mathcal{D}', \mathcal{D}^k(\mathbf{u})$ where $\gamma'' = \gamma\gamma'\delta^9$ (we replace factors of 1/4 with δ). Setting $\rho = 1 - 1/K$, lemma 4.7 implies each $y \in T$ is $((k-3)\sigma/4, 1 - 2\delta^9, \gamma''', 2\ell)$-covered by $\mathcal{D}'''(\mathbf{u}) = \mathcal{D}(\hat{\mathbf{u}})$ for $\hat{\mathbf{u}} \sim_{1-1/K} \mathbf{u}$ where $\gamma''' = \delta^{18K+1}\gamma''$. Finally, since $\delta^5|T_t|/4|S| \geq \delta^7$, lemma 4.6 implies at least $\delta^5|T_t|/2$ vertices in S are $((k+1)\sigma/4, \delta^7, \gamma''', \infty)$-covered by $\mathcal{D}^{k+1}(\mathbf{u}) = \mathbf{u}, \mathcal{D}'''(\mathbf{u})$, where

$$\gamma''' = \delta^{18K+1+3+6k}\gamma/2 \geq 2\delta^{24K(k+1)}$$

Assuming case 2 does not hold for $\mathcal{D}^* = \mathcal{D}^{k+1}$, at least $\delta^5|T_t|/4 \geq \delta^{6(k+1)}|X|$ vertices in S are $((k+1)\sigma/4, \delta^8, \delta^{24K(k+1)}, \ell)$-covered by $\mathcal{D}_{k+1}$.

Finally, note each $\mathcal{D}_k$ consists of a shuffling of $\mathcal{N}^k_{1-1/K}$ with $\mathcal{N}^{k'}_0$ where $k' \leq 6k$ because the expanding case can occur at most $6k$ total times. $\square$

To complete the proof of theorem 2.3, recall M is a $(\sigma, \delta/4)$-uniform-matching-cover of X. Let $1 \leq R \leq \log(n)/\log(1/\delta))$. For $R < 7$, lemma 4.8 implies a typical edge in M has length $\Omega(\sigma/\sqrt{\log(n/\delta)}) = \Omega(R\sigma/\sqrt{\log n})$ since $\log(n \geq \log(1/\delta))$ by assumption. That is, setting $\mathcal{D} = \texttt{Sample}(1, 0)$ suffices.

For $R \geq 7$, set $K = \lfloor R/7 \rfloor$ and $\ell = R\sigma/2^{10}\sqrt{\log(n)}$, so that ℓ satisfies lemma 4.8. Lemma 4.4 implies case 1 of lemma 4.8 can not hold for K, so case 2 must hold. That is, setting $\mathcal{D} = \texttt{Sample}(R, 1 - 1/K)$ suffices.

4.4 Using ± 1 Coins

One might be concerned with issues of precision required for sampling Gaussians. Fortunately, it suffices to approximate them by sampling $\mathbf{w} \in \{\pm 1\}^k$ and returning $\frac{1}{\sqrt{k}} \sum_{i=1}^k \mathbf{w}_i$ for $k = O(\log n)$.

Lemma 4.9. *Suppose that instead of a random Gaussian* $\mathbf{u}$*, we sample a uniform random* ± 1 *matrix* $\mathbf{U} \in \mathbb{R}^{d \times k}$ *and set* $\mathbf{u} = \mathbf{U}\mathbf{1}$*, where* $\mathbf{1} \in \mathbb{R}^k$ *has* $\mathbf{1}_j = 1/\sqrt{k}$ *for all* $j \leq k$*. To sample a* ρ*-correlated copy* $\hat{\mathbf{u}}$*, we sample* $\hat{\mathbf{U}} \in \mathbb{R}^{d \times k}$ *as a* ρ*-correlated copy of* $\mathbf{U}$ *(i.e., each* $\hat{\mathbf{U}}_{ij} = \mathbf{U}_{ij}$ *with probability* ρ *or a random* ± 1 *with probability* $1 - \rho$*) and set* $\hat{\mathbf{u}} = \hat{\mathbf{U}}\mathbf{1}$*. Then, for* $k = O(R^2 \log(1/\delta)) = O(\log n)$*, theorem 2.3 still holds.*

The proof of lemma 4.9 is straightforward. Lemmas 3.3, 4.4, 4.5 all still hold with similar constants even for $k = 1$ (see e.g. [1]), so the only issue is lemma 4.7. For the latter, lemma 4.2 also holds for ρ-correlated ± 1 variables, so the only change needed is in bounding $(\mathbf{v}_x - \mathbf{v}_y) \cdot (\mathbf{u} - \hat{\mathbf{u}})$, which is easily done for $k = O(R^2 \log(1/\delta)) = O(\log(n))$ using Bernstein's inequality(see e.g. [10]). For completeness, we include the details in appendix B.

5 Lower-bound for the Cut-Matching Game

Khandekar, Rao, and Vazirani proposed a primal-dual framework based on the following two-player game game, which proceeds for T rounds[15]. On each round, the cut player chooses a bisection $(S^t, \overline{S^t})$ of the vertices, and the matching player responds with a perfect matching M^t pairing each $x \in S^t$ with some $y \in \overline{S^t}$ The payoff to the cut player is $h(H^T)$, where $H^t = M^1 + \cdots + M^t$. Thus on round t, the cut player aims to choose a cut so that *any* matching response M^t will increase the expansion of H^t.

To see the connection to SPARSEST CUT, suppose the cut player has a strategy that guarantees $h(H^T) \geq T/\kappa$, and consider a matching player that plays as follows. When given a bisection $(S, \overline{S})$, the matching player connects a source s to all $x \in S$ with edges of unit capacity and a sink t to all $y \in \overline{S}$. A simple lemma similar to lemma 3.4 implies that if the min-cut is at most $n/2$, then it has expansion at most one. Otherwise, the added edges are saturated, and assuming all edges have integral capacities, the flow can be pseudo-decomposed into a matching; the matching player responds with that matching. Then, after T rounds, we have either found a cut of expansion one or else routed H^T in G with congestion T. Assuming the cut-player forced $h(H^T) \geq T/\kappa$, scaling down by T yields a feasible flow routing a graph of expansion $1/\kappa$, yielding a κ-approximation.

The following theorems appear in [20].

Theorem 5.1 ([20]). *The cut player has an (efficient) strategy to ensure,*

$$\exp\left(-\lambda_2(\mathcal{L}_{H^t})\right) \leq n \exp\left(\frac{-t}{O(\log n)}\right)$$

In particular, after $T = O(\log^2(n))$, the cut player can ensure $\lambda_2(\mathcal{L}_{H^T}) \geq \Omega(\log n)$, yielding an $O(\log n)$ factor approximation using $O(\log^2 n)$ max-flows.

Theorem 5.2 ([20]). *The matching player can ensure*

$$h(H^t) \leq O\left(\frac{1}{\sqrt{\log n}}\right) \cdot t$$

We prove the following.

Theorem 5.3. *The matching player can ensure*

$$\lambda_2(\mathcal{L}_{H^t}) \leq O\left(\frac{\log \log n}{\log n}\right) \cdot t$$

Theorem 5.3 does not entirely eliminate the possibility of achieving a better approximation in the cut-matching game, and indeed it is known among experts that there *exists* an (inefficient) strategy for the cut-player to ensure $\exp(-h(H^t)) \leq n \exp(-t/O(\sqrt{\log(n)})$[19]. However, theorem 5.3 says that doing so will require certifying expansion via something stronger than $\lambda_2(\mathcal{L}_{H^t})$. For example, one could route another expander flow H' in H^T and certify $h(H^T) \geq \lambda_2(\mathcal{L}_{H'})/2$. Such an approach seems somewhat awkward though, as any such flow might as well have been routed in G directly.

In theorem 5.2, the matching player arbitrarily identifies the vertices of G with a hypercube, and tries to keep the dimension cuts sparse. In particular, it is shown that for any bisection $(S, \overline{S})$, there must always exist a matching that raises the expansion of the average dimension cut by at most $O(1/\sqrt{d})$.

To prove theorem 5.3, we identify the vertices of G arbitrarily with a dense set of points $\mathbf{v}_1, \ldots, \mathbf{v}_n \in \mathbb{R}^d$ on the sphere S^{d-1}, where $d = \Omega(\log(n)/\log\log(n))$. Letting $\mathbf{w}_1, \ldots, \mathbf{w}_d \in \mathbb{R}^n$ be the column vectors of the $n \times d$ matrix with row vectors $\{\mathbf{v}_x\}$, we show that for any bisection $(S, \overline{S})$ there must be a matching that raises the average Rayleigh quotient $\frac{\mathbf{w}_i^T \mathcal{L}_H \mathbf{w}_i}{\mathbf{w}_i^T \mathbf{w}_i}$ by at most $O(1/d)$.

The following lemma is an easy generalization of one in [20].

Lemma 5.4. *Let $\mathbf{v}_1, \ldots, \mathbf{v}^n \in R^d$, and let $\mathbf{w}_1, \ldots, \mathbf{w}_d \in \mathbb{R}^n$ be defined by $\mathbf{w}_{i,x} = \mathbf{v}_{x,i}$. Define,*

$$\psi(t) = \frac{1}{d} \sum_{i=1}^{d} \frac{\mathbf{w}_i^T \mathcal{L}_{H^t} \mathbf{w}_i}{\mathbf{w}_i^T \mathbf{w}_i}$$

If all $\|\mathbf{w}_i\|^2 \geq L > 0$, then,

$$\psi(t) - \psi(t-1) \leq \frac{1}{dL} \sum_{xy \in M^t} \|\mathbf{v}_x - \mathbf{v}_y\|^2$$

Proof.

$$\begin{aligned}
\psi(t) - \psi(t-1) &= \frac{1}{d} \sum_{i=1}^{d} \frac{\mathbf{w}_i^T \mathcal{L}_{M^t} \mathbf{w}_i}{\mathbf{w}_i^T \mathbf{w}_i} \\
&\leq \frac{1}{dL} \sum_{i=1}^{d} \sum_{xy \in M^t} (\mathbf{w}_{i,x} - \mathbf{w}_{i,y})^2 \\
&= \frac{1}{dL} \sum_{xy \in M^t} \|\mathbf{v}_x - \mathbf{v}_y\|^2
\end{aligned}$$

□

If $\mathbf{w}_1, \dots, \mathbf{w}_d$ are as in lemma 5.4 and all orthogonal to the all-1s vector, then $\lambda_2(\mathcal{L}_{H^t}) \leq \psi(t)$; the orthogonality condition is equivalent to $\sum_x \mathbf{v}_x = 0$. Having fixed such an embedding, when presented with a bisection $(S, \overline{S})$, the matching player aims to match points so as to minimize the average distance between matched points. The analysis of [20] shows that for the hypercube embedding $\{-1, 1\}^d$, one can obtain $\psi(t) - \psi(t-1) \leq O(1/\sqrt{d})$. The analysis is not constructive; rather, they use the vertex isoperimetry of the hypercube to establish an upper bound on the value of the matching problem's dual LP, and then conclude a matching achieving that bound exists by strong duality. Their argument also depends on the fact that for the hypercube embedding, the squared distances $\|\mathbf{v}_x - \mathbf{v}_y\|^2$ form a metric.

In fact, the metric assumption is not needed, and there is also no need to apply LP duality. We give a simple proof that large vertex isoperimetry of the embedding implies the simple greedy strategy of iteratively matching closest points works.

Lemma 5.5. *Let $\mathbf{v}_1, \dots, \mathbf{v}_n \in \mathbb{R}^d$ be a set of points such that, for any $S \subseteq \{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ with $|S| \leq n/2$, $|\mathsf{Ball}[S; \sqrt{r}]| \geq (1 + \Omega(1))|S|$, Then, the greedy strategy produces M with,*

$$\sum_{xy \in M} \|\mathbf{v}_x - \mathbf{v}_y\|^2 \leq O(nr)$$

Proof. Starting with $S, \overline{S}$, we pick $x \in S, y \in \overline{S}$ minimizing $\|\mathbf{v}_x - \mathbf{v}_y|^2$, match them, and then remove them. Repeated application of the the isoperimetric condition implies that, for all S with $|S| \leq n/2$, $|\mathsf{Ball}[S; t\sqrt{r}]| \geq \min\{1 + n/2, (1 + \Omega(1))^t |S|\}$. It follows that if two sets A, B have size s, there must be $x \in A$, $y \in B$ with $\|\mathbf{v}_x - \mathbf{v}_y\| \leq 2t\sqrt{r}$ for $t = \lceil \log_{(1+\Omega(1))}(n/2s) \rceil + 1 \leq 2 + O(1)\log(n/2s)$. Then, the total cost of the greedy solution is at most,

$$\begin{aligned}
\sum_{xy \in M} \|\mathbf{v}_x - \mathbf{v}_y\|^2 &\leq O\left(\sum_{s=1}^{n/2} (1 + \log(n/2s))^2 \cdot r \right) \\
&\leq O\left(n + \int_0^{n/2} \log^2(n/2s) \ \mathrm{d}s \right) r \\
&\leq O(nr)
\end{aligned}$$

□

For the case of theorem 5.2, let $\mathbf{v}_x \in \{-1/\sqrt{d}, 1/\sqrt{d}\}^d$ be the hypercube embedding and take $L = n/d$ in lemma 5.4. The vertex isoperimetry of the hypercube implies $r = O(1/\sqrt{d})$ in lemma 5.5, yielding a strategy to ensure $\psi(t) \leq O(nr/dL) \cdot t = O(1/\sqrt{d}) \cdot t$.

To prove theorem 5.3, we choose $\mathbf{v}_x$ as per the following lemma, and take $L = \Omega(n/d)$, $r = O(1/d)$, yielding a strategy to ensure $\psi(t) \leq O(nr/dL) \cdot t = O(1/d) \cdot t$.

Lemma 5.6. *For every d, there exists a set of $n = O(\sqrt{d})^d$ points $\mathbf{v}_1, \dots, \mathbf{v}_n \in S^{d-1}$ such that $\sum_{i=1}^n \mathbf{v}_i = 0$, every $i \leq d$ has $\sum_{x=1}^n \mathbf{v}_{x,i}^2 = \Omega(n/d)$, and for every $S \subseteq [n]$ with $|S| \leq n/2$, $|\mathsf{Ball}[S; O(1/\sqrt{d})]| \geq (1+\Omega(1))|S|$.*

The proof of lemma 5.6 is a straightforward application of a construction of Feige and Schechtman[11], which we include in appendix C

6 Final Remarks

It will be interesting to see if efficient algorithms can be designed for the GENERALIZED SPARSEST CUT problem, where we are given graphs G and H and aim to find a cut $(S, \overline{S})$ minimizing $\frac{\sum_{x \in S, y \in \overline{S}} G_{xy}}{\sum_{x \in S, y \in \overline{S}} H_{xy}}$ (when H is the complete graph, the problem is essentially the regular SPARSEST CUT problem, up to a factor of two). The results of [6] imply an $O(\sqrt{\log n} \log \log n)$-approximation can be found by rounding a SDP similar to (1), but to the best of our knowledge no efficient algorithms have been designed to approximately solve that SDP.

The boosting step in our proof of theorem 2.3 crucially depends on use of the *noise operator*. Many hardness of approximation reductions for CSPs also make use of that operator in their soundness analysis; what is the connection between how it is used in each case?

Another question concerns the relation between the expander flow SDP and the original "stronger" SDP proposed by Goemans. Constructing integrality gaps for the latter is a notoriously hard problem. Might it be any easier to construct them for (1)? If not, can one always "round" an embedding for the dual of (1) to an embedding satisfying the triangle inequality constraints of Goemans' program?

Acknowledgement

We thank Umesh Vazirani and Satish Rao for helpful discussions, Ryan O'Donnell for suggesting [9, 18] to prove lemma 4.2, and James Lee for suggesting [11] to prove lemma 5.6.

References

[1] Dimitris Achlioptas. Database-friendly random projections: Johnson-lindenstrauss with binary coins. *J. Comput. Syst. Sci.*, 66(4):671–687, 2003.

[2] Noga Alon and V. D. Milman. λ_1, isoperimetric inequalities for graphs, and superconcentrators. *J. Comb. Theory, Ser. B*, 38(1):73–88, 1985.

[3] Sanjeev Arora, Elad Hazan, and Satyen Kale. $O(\sqrt{\log n})$ 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*, pages 238–247, Washington, DC, USA, 2004. IEEE Computer Society.

[4] Sanjeev Arora, Elad Hazan, and Satyen Kale. The multiplicative weights update method: a meta algorithm and applications. Technical report, Princeton University, 2005.

[5] Sanjeev Arora and Satyen Kale. A combinatorial, primal-dual approach to semidefinite programs. In *STOC '07: Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 227–236, New York, NY, USA, 2007. ACM.

[6] Sanjeev Arora, James R. Lee, and Assaf Naor. Euclidean distortion and the sparsest cut. In *STOC '05: Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 553–562, New York, NY, USA, 2005. ACM.

[7] Sanjeev Arora, Satish Rao, and Umesh Vazirani. Expander flows, geometric embeddings and graph partitioning. In *STOC '04: Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 222–231, New York, NY, USA, 2004. ACM.

[8] András A. Benczúr and David R. Karger. Approximating s-t minimum cuts in $O(n^2)$ time. In *STOC '96: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 47–55, New York, NY, USA, 1996. ACM.

[9] Christer Borell. Positivity improving operators and hypercontractivity. *Mathematische Zeitschrift*, 180:225–234, 1982.

[10] Stéphane Boucheron, Gábor Lugosi, and Olivier Bousquet. Concentration inequalities. In *Advanced Lectures on Machine Learning*, pages 208–240. Springer, 2003.

[11] Uriel Feige and Gideon Schechtman. On the optimality of the random hyperplane rounding technique for max cut. *Random Struct. Algorithms*, 20(3):403–440, 2002.

[12] Lisa K. Fleischer. Approximating fractional multicommodity flow independent of the number of commodities. *SIAM Journal on Discrete Mathematics*, 13:505–520, 2000.

[13] Yoav Freund and Robert E. Schapire. Adaptive game playing using multiplicative weights. *Games and Economic Behavior*, 29(1-2):79–103, October 1999.

[14] Andrew V. Goldberg and Satish Rao. Beyond the flow decomposition barrier. *J. ACM*, 45(5):783–797, 1998.

[15] Rohit Khandekar, Satish Rao, and Umesh Vazirani. Graph partitioning using single commodity flows. In *STOC '06: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 385–390, New York, NY, USA, 2006. ACM.

[16] James R. Lee. On distance scales, embeddings, and efficient relaxations of the cut cone. In *SODA '05: Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 92–101, Philadelphia, PA, USA, 2005. Society for Industrial and Applied Mathematics.

[17] Jiri Matousek. *Lectures on Discrete Geometry.* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2002.

[18] Elchanan Mossel, Oded Regev, Jeffrey E. Steif, and Benny Sudakov. Non-interactive correlation distillation, inhomogeneous markov chains, and the reverse bonami-beckner inequality. *Israel Journal of Mathematics*, 154, 2006.

[19] Lorenzo Orecchia. personal communication, 2009.

[20] Lorenzo Orecchia, Leonard J. Schulman, Umesh V. Vazirani, and Nisheeth K. Vishnoi. On partitioning graphs via single commodity flows. In *STOC '08: Proceedings of the 40th annual ACM symposium on Theory of computing*, pages 461–470, New York, NY, USA, 2008. ACM.

[21] Serge A. Plotkin, David B. Shmoys, and Eva Tardos. Fast approximation algorithms for fractional packing and covering problems. *Mathematics of Operations Research*, 20:257–301, 1995.

[22] Daniel D. Sleator and Robert Endre Tarjan. A data structure for dynamic trees. *J. Comput. Syst. Sci.*, 26(3):362–391, 1983.

A Proof of Lemma 4.2

For $f : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$, let $\|f\|_p = \mathbf{E}[f^p]^{1/p}$, where the expectation is over the multivariate standard normal distribution. For $x \in \mathbb{R}^d$, we write $y \sim_\rho x$ for a ρ-correlated copy of u. The *Ornstein-Uhlenback operator* is defined by,

$$T_\rho f(x) = \mathbf{E}_{y \sim_\rho x}[f(y)]$$

Theorem A.1 (Borell[9])**.** *Let $f : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ and $-\infty < q \leq p \leq 1$. If $0 \leq \rho^2 \leq (1-p)/(1-q)$, then*

$$\|T_\rho f\|_q \geq \|f\|_p \qquad \text{for } 0 \leq \rho^2 \leq (1-p)/(1-q)$$

By a change of variables, lemma 4.2 is equivalent to,

$$\mathbf{Pr}_{\mathbf{u}}[\mathbf{Pr}_{\hat{\mathbf{u}}}[\hat{\mathbf{u}} \in A] < \tau] < \frac{\tau^{1-\rho}}{\delta}$$

Let f indicate A, and set $p = 1-\rho, q = 1 - 1/\rho$. Note $q < 0 < p \leq 1$ satisfy theorem A.1, so

$$\|T_\rho f\|_q \geq \|f\|_p = \delta^{1/p}$$

Then, $\mathbf{Pr}_{\hat{\mathbf{u}} \sim_\rho \mathbf{u}}[\hat{\mathbf{u}} \in A] = T_\rho f(\mathbf{u})$, and we have,

$$\begin{aligned}
\mathbf{Pr}[T_\rho f < \tau] &= \mathbf{Pr}[(T_\rho f)^q > \tau^q] \\
&< \|(T_\rho f)\|_q^q \tau^{-q} \\
&\leq \delta^{q/p} \tau^{-q} \\
&= \left(\frac{\tau^{1-\rho}}{\delta}\right)^{1/\rho}
\end{aligned}$$

For $\tau^{1-\rho}/\delta \leq 1$, raising the last line to ρ can't decrease its value. In the other case, the result is trivial.

B Proof of Lemma 4.9

Lemma 3.3 only uses the fact that for a vector $\mathbf{v}$ and standard normal $\mathbf{u}$, $(\mathbf{u} \cdot \mathbf{v})^2 \geq \Omega(\|\mathbf{v}\|^2)$ with probability $\Omega(1)$. That property still holds.

Lemma B.1. *Let $\mathbf{U} \in \mathbb{R}^{d \times k}$ be a uniform random ± 1 matrix, and let $\mathbf{v} \in \mathbb{R}^d$ be a vector. Then,*

$$\mathbf{Pr}\left[(\mathbf{v} \cdot \mathbf{U}\mathbf{1})^2 \geq \|\mathbf{v}\|^2/4\right] \geq 1/5$$

Proof. It suffices to consider a unit vector $\mathbf{v}$. Let $Z = \mathbf{v} \cdot \mathbf{U1}$. Then,

$$\mathbf{E}[Z^2] = \mathbf{E}\left[\left(\sum_{i \le d, j \le k} \mathbf{v}_i \frac{\mathbf{U}_{ij}}{\sqrt{k}}\right)^2\right] = \sum_{i_1,i_2,j_1,j_2} \mathbf{v}_{i_1}\mathbf{v}_{i_2} \frac{\mathbf{E}\left[\mathbf{U}_{i_1 j_1}\mathbf{U}_{i_2 j_2}\right]}{k} = \sum_{i,j} \mathbf{v}_i^2/k = \|\mathbf{v}\|^2 = 1$$

$$\mathbf{E}[Z^4] = \sum_{i_1,\ldots,i_4,j_1,\ldots,j_4} \mathbf{v}_{i_1} \cdots \mathbf{v}_{i_4} \frac{\mathbf{E}\left[\mathbf{U}_{i_1 j_1} \cdots \mathbf{U}_{i_4 j_4}\right]}{k^2} \le 3 \sum_{i_1,j_1,i_2,j_2} \mathbf{v}_{i_1}^2 \mathbf{v}_{i_2}^2/k^2 = 3\|\mathbf{v}\|^4 = 3$$

Then, for any $t \le 1/\lambda$, we have,

$$\mathbf{Pr}\left[Z^2 < \lambda\right] \le \mathbf{Pr}\left[\left(1 - tZ^2\right)^2 > (1 - t\lambda)^2\right] < \frac{\mathbf{E}\left[\left(1 - tZ^2\right)^2\right]}{(1 - t\lambda)^2} = \frac{1 - 2t\mathbf{E}[Z^2] + t^2\mathbf{E}[Z^4]}{(1 - t\lambda)^2}$$

Taking $t = \lambda = 1/4$ yields,

$$\mathbf{Pr}\left[Z^2 < 1/4\right] < \frac{1 - 1/2 + 3/16}{(15/16)^2} < 1/5$$

□

The remaining lemmas require a Gaussian-like bound on stretch; for that, we'll use the following theorem.

Theorem B.2 (Bernstein's Inequality)**.** *Let $X_1, \ldots, X_n$ be independent random variables with $\mathbf{E}[X_i] = 0$ and $X_i \le 1$. Let $\sigma^2 = \sum_{i=1}^n \mathbf{E}[X_i^2]$. Then, for any $t > 0$,*

$$\mathbf{Pr}\left[\sum_{i=1}^n X_i > t\sigma\right] \le \exp\left(\frac{-t^2}{2 + t/3\sigma}\right)$$

The next lemma says that if k is large enough, we can obtain Gaussian-like bounds on stretch. Note that when $\rho = 0$ much better bounds are possible, in that even $k = 1$ works (see [1]).

Lemma B.3. *Let $\mathbf{U} \in \mathbb{R}^{d \times k}$ be an arbitrary ± 1 matrix, and let $\hat{\mathbf{U}} \sim_\rho \mathbf{U}$ be a ρ-correlated copy of $\mathbf{U}$. Then, for any vector $\mathbf{v}$, and any $0 < t \le \sqrt{k(1 - \rho^2)}$,*

$$\mathbf{Pr}\left[\mathbf{v} \cdot \hat{\mathbf{U}}\mathbf{1} > \rho(\mathbf{v} \cdot \mathbf{U1}) + t\sqrt{1 - \rho^2}\|\mathbf{v}\|\right] \le e^{-t^2/3}$$

$$\mathbf{Pr}\left[\mathbf{v} \cdot \hat{\mathbf{U}}\mathbf{1} < \rho(\mathbf{v} \cdot \mathbf{U1}) - t\sqrt{1 - \rho^2}\|\mathbf{v}\|\right] \le e^{-t^2/3}$$

Proof. It suffices to consider a unit vector $\mathbf{v}$. For each $i \le d$, $j \le k$, let $Z_{ij} = \mathbf{v}_i(\hat{\mathbf{u}}_{ij} - \rho\mathbf{U}_{ij})/2$, so that we have $\mathbf{E}[Z_{ij}] = 0$, $|Z_{ij}| \le 1$, and $\mathbf{E}[Z_{ij}^2] = (1 - \rho^2)\mathbf{v}_i^2/4$. Note that,

$$\mathbf{v} \cdot \hat{\mathbf{U}}\mathbf{1} = \rho(\mathbf{v} \cdot \hat{\mathbf{U}}\mathbf{1}) + \frac{2}{\sqrt{k}} \sum_{i \le d, j \le k} Z_{ij}$$

Applying theorem B.2 with $\sigma^2 = k(1 - \rho^2)\|\mathbf{v}\|^2/4 = k(1 - \rho^2)/4$, we have

$$\mathbf{Pr}\left[\sum_{i,j} Z_{ij} > t\sigma\right] \le \exp\left(\frac{-t^2}{2 + t/3\sigma}\right) \le e^{-t^2/3}$$

proving the first part. The second part follows by applying the same argument to $-Z_{ij}$. □

For lemmas 4.4 and 4.5, we use $\rho = 0$ and $t = O(\sqrt{\log(1/\delta)})$, so $k = O(\log(1/\delta))$ suffices. For lemma 4.7, we use $\rho = 1 - 1/K$ and $t = O(\sqrt{K \log(1/\delta)})$, so $k = O(K^2 \log(1/\delta))$ suffices. Also, lemma 4.2 holds for the uniform measure on the hypercube, as Borell's theorem also holds for $f : \{-1, +1\}^n \to \mathbb{R}_{\ge 0}$.

C Proof of Lemma 5.6

Lemma C.1 (Feige, Schechtman [11]). *For each $0 < \gamma < \pi/2$, the sphere S^{d-1} can be partitioned into $n = (O(1)/\gamma)^d$ equal volume cells, each of diameter at most γ.*

Apply lemma C.1 with $\gamma = 1/\sqrt{d}$, yielding cells $C_1, \dots, C_n$ with $n = \exp(O(d \log d))$. Let V be a set of n arbitrary points, each from a distinct cell; for convenience, let us choose V so that $V \cap (-V) = \emptyset$.

Claim C.2. *If $A \subseteq S^{d-1}$ has $\mu(A) \geq \alpha$, then $|\mathsf{Ball}(A;\gamma) \cap V| \geq \alpha n$; if $A \subseteq V$ has $|A| \geq \alpha n$, then $\mu(\mathsf{Ball}(A;\gamma)) \geq \alpha$.*

Proof. For the first direction, if $\mu(A) \geq \alpha$, A intersects at least αn of the cells, so $\mathsf{Ball}[A;\gamma]$ contains at least αn cells, and hence at least αn elements of V. For the second, if $A \subseteq V$ has size at least αn, then $\mathsf{Ball}[A;\gamma]$ contains at least αn cells, so $\mu(\mathsf{Ball}[A;\gamma]) \geq \alpha$. □

Claim C.3. *For every $i \leq d$, $\sum_{\mathbf{v} \in V} \mathbf{v}_i^2 \geq \Omega(n/d)$.*

Proof. Let $A = \{x \in S^{d-1} : x_i \geq \sqrt{2/d}\}$. By bounds on the measure of spherical caps (see e.g. [17]), $\mu(A) \geq 1/12$. Using claim C.2, $|\mathsf{Ball}(A;\gamma) \cap V| \geq (1/12)n$. Then, since $x_i \geq \sqrt{2/d} - \gamma \geq \sqrt{1/8d}$ for all $x \in \mathsf{Ball}(A;\gamma)$, we have $\sum_{\mathbf{v} \in V} \mathbf{v}_i^2 \geq (1/12)n(1/8d) \geq \Omega(n/d)$. □

Claim C.4. *For every $A \subseteq V$ with $|A| \leq n/2$, $|\mathsf{Ball}[A; O(1/\sqrt{d})] \cap X| \geq (1 + 1/12)|A|$.*

Proof. Let $A \subseteq V$ have $|A| \leq n/2$, and set $A_1 = \mathsf{Ball}[A;\gamma], A_2 = \mathsf{Ball}[A_1; 4/\sqrt{d}], A_3 = \mathsf{Ball}[A_2;\gamma]$; the goal is to show $|A_3 \cap V| \geq (1 + 1/12)|A|$. Note claim C.2 ensures $\mu(A_1) \geq |A|/n$. If $\mu(A_1) \geq (1 + 1/12)/2$, then $\mu(A_2) \geq (1 + 1/12)|A|/n$. Otherwise, by the isoperimetric inequality on the sphere (see e.g. [17]), $\mu(A_2) \geq (1 + 1/12)\mu(A_1) \geq (1 + 1/12)|A|/n$. By by claim C.2, $|A_3 \cap V| \geq (1 + 1/12)|A|$. □

Now to prove lemma 5.6, we let $V' = V \cup -V$. Clearly $\sum_{\mathbf{v} \in V'} \mathbf{v} = 0$, and claim C.3 still applies to V', so it remains only to argue claim C.4 still holds for V'. Let $A \subseteq V'$ have $|A| \leq |V'|/2 = n$. Let $A = A_+ \cup A_-$ where $A_+ \subseteq V$ and $A_- \subseteq -V$, and suppose $|A_+| \leq |A_-|$ (in the other case, an analogous argument applies). We consider two cases. First, if $|A_+| \leq |A_-|/2$, then $\mu(\mathsf{Ball}[A_-;\gamma]) \geq |A_-|/n$, implying $|\mathsf{Ball}[A_-; 2\gamma] \cap V| \geq |A_-|/n$. Therefore, $|\mathsf{Ball}[A; 2\gamma] \cap V'| \geq 2|A_-| \geq (4/3)|A|$. Otherwise, $|A_+| \leq n/2$ and $|A_+| \geq |A|/3$, so claim C.4 implies $|\mathsf{Ball}[A_+; O(1/\sqrt{d})] \cap V| \geq (1 + 1/12)|A_+|$. Therefore, $|\mathsf{Ball}[A; O(1/\sqrt{d})] \cap V'| \geq (1 + 1/12)|A_+| + |A_-| \geq (1 + 1/36)|A|$.