On the Hardness of Approximating Multicut and Sparsest-Cut*

Shuchi Chawla[†] Robert Krauthgamer[‡] Ravi Kumar[§] Yuval Rabani[¶]
D. Sivakumar[‖]

Abstract

We show that the Multicut, Sparsest-Cut, and Min-2CNF≡ Deletion problems are NP-hard to approximate within every constant factor, assuming the Unique Games Conjecture of Khot [STOC, 2002]. A quantitatively stronger version of the conjecture implies an inapproximability factor of $\Omega(\sqrt{\log\log n})$.

1 Introduction

In the Multicut problem the input is an undirected graph $G = (V, E)$ on $n = |V|$ vertices together with k pairs of vertices $\{s_i, t_i\}_{i=1}^k$, called *demand pairs*, and the goal is to find a minimum-size subset of the edges $M \subseteq E$ whose removal disconnects all the demand pairs, i.e., in the subgraph $(V, E \setminus M)$ every s_i is disconnected from its corresponding vertex t_i. In the weighted version of this problem, the input also specifies a positive cost $c(e)$ for each edge $e \in E$ and the goal is to find a subset of the edges $M \subseteq E$ whose total cost $c(M) = \sum_{e \in M} c(e)$ is minimal. This problem is known to be APX-hard [DJP+94].

We prove that if the Unique Games Conjecture of Khot [Kho02] is true, then for every constant $L > 0$ it is NP-hard to approximate Multicut within factor L. If a quantitatively stronger version of the conjecture is true, then Multicut is NP-hard to approximate within a factor of $\Omega(\sqrt{\log\log n})$.

Our methods also yield similar bounds for Sparsest-Cut and Min-2CNF≡ Deletion. The Sparsest-Cut problem has the same input as Multicut, but the goal is to find a subset of the edges $M \subseteq E$ that minimizes the ratio of $|M|$ (in the weighted version, the total cost of M) to the number of demand pairs that are disconnected in $(V, E \setminus M)$.[1] Since Sparsest-Cut is not

*A preliminary version of this paper appeared in Proceedings of 20th Annual IEEE Conference on Computational Complexity (CCC 2005).

[†]Computer Science Department, Carnegie Mellon University, Pittsburgh, PA 15213, USA. Part of this work was done while the author was visiting IBM Almaden Research Center. Email: `shuchi@cs.cmu.edu`

[‡]IBM Almaden Research Center, 650 Harry Road, San Jose, CA 95120, USA. Email: `robi@almaden.ibm.com`

[§]Yahoo! Research, 701 First Avenue, Sunnyvale, CA 94089. This work was done while the author was at the IBM Almaden Research Center. Email: `ravikumar@yahoo-inc.com`

[¶]Computer Science Department, Technion—Israel Institute of Technology, Haifa 32000, Israel. Part of this work was done while the author was on sabbatical leave at Cornell University, while visiting IBM Almaden Research Center, and while visiting the Institute for Pure and Applied Mathematics at UCLA. Research at the Technion supported in part by ISF grant number 52/03 and BSF grant number 02-00282. Email: `rabani@cs.technion.ac.il`

[‖]Google, Inc., 1600 Amphitheatre Parkway, Mountain View, CA 94043. This work was done while the author was at the IBM Almaden Research Center. Email: `siva@google.com`

[1]In general, the demand pairs may have positive weights (demands), but for our purpose of inapproximability results, it clearly suffices to consider the more restricted definition above. Our hardness results for Sparsest-Cut do not apply to the special case of uniform-demand, in which every pair of vertices forms a demand pair.

known to be APX-hard, our result gives the first indication that this problem might be hard to approximate. In the Min-2CNF≡ Deletion problem the input is a weighted set of clauses on n variables, each clause of the form $(x \equiv y)$, where x and y are literals, and the goal is to find a Boolean assignment to the variables minimizing the total weight of unsatisfied clauses.[2] Our results immediately extend also to the Correlation Clustering problem [BBC04, CGW03, DI03, EF03] of minimizing disagreements in a weighted graph, since the approximability of this problem is known to be equivalent to within constant factors to that of Multicut in weighted graphs [CGW03, EF03].

1.1 Known results on Multicut, Sparsest-Cut, and Min-2CNF≡ Deletion

Multicut and Sparsest-Cut are fundamental combinatorial optimization problems, with connections to multicommodity flow, edge expansion, and metric embeddings. Both problems can be approximated to within $O(\log k)$ factors through linear programming (LP) relaxations [LR88, GVY96, AR98, LLR95]. These bounds match, up to constant factors, the lower bounds on the integrality gaps of the corresponding relaxations [LR88, GVY96]. Min-2CNF≡ Deletion can also be approximated to within an $O(\log n)$ factor, as implied by the results of Klein et al. [KARR90] (see also [Vaz01, Section 20.4]), who give an approximation-preserving reduction from this problem to Multicut. Recently, improved approximation algorithms for the Sparsest-Cut problem have been developed using a semidefinite programming (SDP) relaxation [ARV04, CGR05, ALN05]. This started with the ground-breaking $O(\sqrt{\log n})$-approximation of Arora, Rao, and Vazirani [ARV04] for the uniform demands case, and the best approximation factor currently known for general demands is $O(\sqrt{\log k}\log\log k)$ [ALN05]. Agarwal et al. [ACMM05] extended these techniques to obtain an $O(\sqrt{\log n})$ approximation for the Min-2CNF≡ Deletion problem. The obvious modification of the semidefinite program used for Sparsest-Cut to solve Multicut was recently shown to have an integrality ratio of $\Omega(\log k)$ [ACMM05, Section 6], which matches, up to constant factors, the approximation factor and integrality gap of previously analyzed linear programming relaxations for this problem.

On the hardness side, it is known that Multicut is APX-hard [DJP+94], i.e., there exists a constant $c > 1$, such that it is NP-hard to approximate Multicut to within a factor smaller than c. It should be noted that this hardness of approximation holds even for $k = 3$, and that the value of c is not specified therein, but it is certainly smaller than 2. The Min-2CNF≡ Deletion problem is also known to be APX-hard, as follows, e.g., from the hardness of approximating linear equations modulo 2 [Hås01].

Assuming the Unique Games Conjecture, Khot [Kho02, Theorem 3] essentially obtained an arbitrarily large constant-factor hardness for Min-2CNF≡ Deletion, and this implies, using the aforementioned reduction of [KARR90], a similar hardness factor for Multicut. These results are not explicitly noted in [Kho02], and are weaker than our results in several respects. First, our quantitative bounds are better; thus if a stronger, yet almost as plausible, version of this conjecture is true, then our lower bound on the approximation factor improves to $L = \Omega(\sqrt{\log\log n})$, compared with the roughly $\Omega((\log\log n)^{1/4})$ hardness that can be inferred from [Kho02]; this can be viewed as progress towards proving tight inapproximability results for Multicut. Second, by qualitatively strengthening our Multicut result to a bicriteria version of the problem, we extend our hardness results to the Sparsest-Cut problem. It is unclear whether Khot's reduction similarly leads to a hardness result for Sparsest-Cut. Finally, our proof is simpler (both the reduction and its analysis), and makes direct connections to cuts (in a hypercube), and thus may prove useful in further investigation of such questions.

[2] Note that the constraints in Min-2CNF≡ Deletion are restricted to equality (and effectively non-equality) constraints.

For SPARSEST-CUT, no hardness of approximation result was previously known. Independently of our work, Khot and Vishnoi [KV05] have recently used a different construction to show an arbitrarily large constant factor hardness for SPARSEST-CUT assuming the Unique Games Conjecture. Their hardness factor could, in principle, be pushed to $(\log\log n)^{1/4-o(1)}$, assuming a stronger quantitative version of the conjecture. Additionally, they prove a lower bound of $(\log\log n)^{1/4-o(1)}$ on the integrality ratio of the semidefinite programming relaxations used in the recent approximation algorithms for SPARSEST-CUT. This last lower bound was further improved to $\Omega(\log\log n)$ in [KR06], using in part ideas from the current paper.

1.2 The Unique Games Conjecture

Unique 2-prover game is the following problem. The input is a bipartite graph $G_Q = (Q, E_Q)$, where each side $p = 1, 2$ contains $n = |Q|/2$ vertices denoted $q_1^p, \cdots, q_n^p$, and represents n possible questions to prover p. In addition, the input contains for each edge $(q_i^1, q_j^2) \in E_Q$ a non-negative weight $w(q_i^1, q_j^2)$, abbreviated w_{ij}. These edges will be called *question edges*, to distinguish them from edges in the MULTICUT instance. Each question to a prover is associated with a set of d distinct answers, denoted by $[d] = \{1, \ldots, d\}$. The input also contains, for every question edge $(q_i^1, q_j^2) \in E_Q$, a bijection $b_{ij} : [d] \to [d]$, which maps every answer for question q_i^1 to a distinct answer for question q_j^2.

A *solution* A to the 2-prover game consists of an answer $A_i^p \in [d]$ for each question q_i^p (i.e., a sequence $\{A_i^p\}$ over all $p \in [2]$ and $i \in [n]$). The solution is said to satisfy a question edge $(q_i^1, q_j^2) \in E_Q$ if the answers A_i^1 and A_j^2 agree under their bijection, i.e., $A_j^2 = b_{ij}(A_i^1)$. We assume that the total weight of all the edges in E_Q is 1 (by normalization). The *value of a solution* is the total weight of all the edges satisfied by the solution. The *value of the game* is the maximum value achievable by any solution to the game.

Conjecture 1.1 (Unique Games [Kho02]). *For every fixed $\eta, \delta > 0$ there exists $d = d(\eta, \delta)$ such that it is NP-hard to determine whether a unique 2-prover game with answer set size d has value at least $1 - \eta$ or at most δ.*

We will also consider stronger versions of the Unique Games Conjecture in which η, δ, and d are functions of n. Specifically, we will consider versions with $\eta \le O(1/\sqrt{\log\log n})$, $\delta \le 1/(\log n)^{\Omega(\eta)}$, and $d = d(\eta, \delta) \le O(\log n)$. The reason for the latter upper bound is that our construction size is exponential in d.

Plausibility of the conjecture and stronger versions of it. The Unique Games Conjecture has been used to show optimal inapproximability results for VERTEX COVER [KR03] and MAX-CUT [KKMO04, MOO05]. Proving the conjecture using current techniques appears to be quite hard. In particular, the asserted NP-hardness is much stronger than what we can obtain via standard constructions using the PCP theorem [AS98, ALM+98] and the parallel repetition theorem [Raz98], two deep results in computational complexity.

Although the conjecture seems difficult to prove in general, some special cases are well-understood. In particular, if at all the Unique Games Conjecture is true (and assuming P $\neq$ NP), then necessarily $d \ge \max\{1/\eta^{1/10}, 1/\delta\}$. This follows from a rounding procedure for a semidefinite programming relaxation presented in [Kho02]. On the other hand, Feige and Reichman [FR04] showed that for every constant $L > 0$ there exists a constant $\delta > 0$, such that it is NP-hard to distinguish whether a unique 2-prover game (with $d = d(L, \delta)$) has value at least $L\delta$ or at most δ; this result falls short of the Unique Games Conjecture in that $L\delta$ is bounded away from 1.

Stronger versions of the conjecture in which d, η, and δ are functions of n have also been considered. Trevisan [Tre05] and Gupta and Talwar [GT06] recently developed approximation algorithms for solving instances of Unique Games where $\eta(n)$ is a sufficiently small function of n (based on an SDP and an LP relaxation respectively, different from the one used in [Kho02]). These algorithms imply that a stronger version of the Unique Games Conjecture can only be true if $\eta(n) = \Omega(1/\log n)$ (assuming $\mathrm{P} \neq \mathrm{NP}$).

Very recently, Charikar, Makarychev, and Makarychev [CMM06] improved upon the rounding algorithm of Khot [Kho02] to obtain better approximation algorithms for Unique Games. Their results imply that for Unique Games Conjecture to be true, we must have $\eta \geq \Omega(1/\log d)$ and $\delta \geq d^{-\Omega(\eta)}$. There are two pieces of evidence suggesting that, if the Unique Games Conjecture is true at all, then these bounds might be tight, i.e. Unique Games might in fact be hard for suitable $d = \Theta(\log n)$, and every $\delta \geq d^{-\Omega(\eta)}$ and $1/\log \ll d \leq \eta < 1/2$. First, these bounds nearly match the integrality gap of [KV05] for the natural SDP relaxation of Unique Games. Second, a recent version of [KKMO04] shows (see Corollaries 13 and 15 of its ECCC version) that the standard Unique Games Conjecture implies hardness of Unique Games instances with the above tradeoff between parameters (namely, $\delta = d^{-\eta/(2-\eta)}$), for every $\eta > 0$ that is fixed (independent of n) and d is larger than some $d_0(\eta)$ that is unspecified.

Our $\Omega(\sqrt{\log\log n})$ hardness result (Corollary 1.4 below) requires $\eta \leq O(1/\sqrt{\log\log n})$, $\delta \leq 1/(\log n)^{\Omega(\eta)}$, and $d = d(\eta,\delta) \leq O(\log n)$, which is still plausible by these results. Note that this setting of parameters is similar to, but stronger than, those in the aforementioned result of [KKMO04], because the parameters do depend on n.

1.3 Our results

We prove the following hardness of approximation for Multicut, Sparsest-Cut, and Min-2CNF≡ Deletion based on the Unique Games Conjecture.

Theorem 1.2. *Suppose that for $\eta = \eta(n)$, $\delta = \delta(n)$, and $d = d(\eta,\delta) \leq O(\log n)$, it is NP-hard to determine whether a unique 2-prover game with $|Q| = 2n$ vertices and answer set size d has value at least $1 - \eta(n)$ or at most $\delta(n)$. Then there exist constants $c_1, c_2 > 0$ such that it is NP-hard to approximate* Multicut, Sparsest-Cut, *and* Min-2CNF≡ Deletion *to within factor $L(n) = c_1 \min\{\frac{1}{\eta(n^{c_2})}, \log\frac{1}{\delta(n^{c_2})}\}$.*

This theorem immediately implies the following two specific hardness results.

Corollary 1.3. *The Unique Games Conjecture implies that, for every constant $L > 0$, it is NP-hard to approximate* Multicut, Sparsest-Cut, *and* Min-2CNF≡ Deletion *to within factor L.*

Corollary 1.4. *A stronger version of the Unique Games Conjecture in which $\eta \leq O(1/\sqrt{\log\log n})$, $\delta \leq 1/(\log n)^{\Omega(\eta)}$, and $d = d(\eta,\delta) \leq O(\log n)$, implies that for some fixed $c > 0$, it is NP-hard to approximate* Multicut, Sparsest-Cut, *and* Min-2CNF≡ Deletion *to within factor $c\sqrt{\log\log n}$.*

For Sparsest-Cut, our hardness results hold only for the search version (in which the algorithm needs to produce a cutset, i.e., a subset of edges, and not only its value), since our proof employs a Cook reduction. As noted before, a similar (but slightly weaker) bound is proved independently in [KV05].

Remark. The conference version of our paper [CKK$^+$05] presented a different bound than that of Theorem 1.2, and argued that an $\Omega(\log\log n)$ hardness result follows from a certain strong version of the Unique Games Conjecture, that has since been proven false by Charikar et al. [CMM06]. Consequently, the current version of the paper is different in two respects. First, we use a different strong version of the Unique Games Conjecture that is still plausibly true by known evidence (and is in fact weaker than the one used in [CKK$^+$05]). Second, the bound presented here in Theorem 1.2 is better, since in [CKK$^+$05] the dependence of $L(n)$ on η was $c_1(\log \frac{1}{\eta(n^{c_2})})$. Interestingly, the two versions employ exactly the same reduction, but the analysis is different (and perhaps simpler), as the current version uses Friedgut's Junta Theorem [Fri98] rather than a theorem of Kahn, Kalai, and Linial [KKL88]. This improvement was also motivated, in part, by the integrality ratio of [KV05] for Unique Games, which suggests a significant asymmetry between $\eta(n)$ and $\delta(n)$.

1.4 Preliminaries

Regular Unique Games. A unique 2-prover game is called *regular* if the total weight of question edges incident at any single vertex is the same, i.e., $1/n$, for every vertex in Q. We now show that we can assume without loss of generality that the graph in the Unique Games Conjecture is regular. For simplicity, we state this only for fixed η and δ. A similar result holds when they depend on n, because we increase the input size by no more than a polynomial factor, and increase η and δ by no more than a constant factor.

Lemma 1.5. *The Unique Games Conjecture implies that for every fixed $\eta, \delta > 0$, there exists $d = d(\eta, \delta)$ such that it is NP-hard to decide if a regular unique 2-prover game with n vertices and d answers for each vertex, has value at least $1 - \eta$ or at most δ.*

The proof is given in Appendix A, and is based on an argument of Khot and Regev [KR03, Lemma 3.3]. The relationship between d, η, δ and n is important for our purposes. We thus point out that this argument changes the parameters δ and η by at most a factor of 2, increases the size of the instance by at most a polynomial factor in n, $1/\eta$ and $1/\delta$, and does not change d. This is acceptable in the setting of Theorem 1.2 as well as Corollary 1.4 — the requirements on d, η and δ as functions of n are maintained and only the unspecified constants therein are affected.

Bicriteria Multicut. Our proof for the hardness of approximating Sparsest-Cut relies on a generalization of Multicut, where the solution M is required to cut only a certain fraction of the demand pairs. For a given graph $G = (V, E)$, a subset of the edges $M \subseteq E$ will be called a *cutset* of the graph. A cutset whose removal disconnects all the demand pairs is a *multicut*.

An algorithm is called an (α, β)-bicriteria approximation for Multicut if, for every input instance, the algorithm outputs a cutset M that disconnects at least α fraction of the demands and has cost at most β times that of the optimum multicut. In other words, if M^* is a least cost cutset that disconnects all the k demand pairs, then M disconnects at least αk demand pairs and $c(M) \leq \beta \cdot c(M^*)$.

Hypercubes, dimension cuts, and antipodal vertices. As usual, the *d-dimensional hypercube* (in short a *d-cube*) is the graph $C = (V_C, E_C)$ with the vertex set $V_C = \{0,1\}^d$, and an edge $(u, v) \in E_C$ for every two vertices $u, v \in \{0,1\}^d$ that differ in exactly one dimension (coordinate). An edge (u, v) is called a *dimension-a edge*, for $a \in [d]$, if u and v differ in dimension a, i.e., $u \oplus v = 1_a$ where 1_a is a unit vector along dimension a. The set of all the dimension-a edges in the hypercube is called the *dimension-a cut* in the hypercube; a *dimension cut* is a dimension-a cut for some dimension a. The *antipode* of a vertex u is the (unique) vertex $\overline{u}$ all of whose coordinates are different from those of u, i.e., $\overline{u} = u \oplus \mathbf{1}$ where $\mathbf{1}$ is the vector with 1 in every coordinate. Notice that v is the antipode of u if and only if u is the antipode of v; thus, $\{u, \overline{u}\}$ form an *antipodal pair.* The following simple fact will be key in our proof.

Fact 1.6. *In every hypercube, a single dimension cut disconnects every antipodal pair.*

Organization. In Section 2 we prove the part of Theorem 1.2 regarding the MULTICUT problem; our proof will actually hold for bicriteria approximation for MULTICUT. We will then show in Section 3 that this stronger result yields a similar hardness of approximation for SPARSEST-CUT. Finally, in Section 4, we modify the reduction to obtain a hardness of approximation for MIN-2CNF≡ DELETION.

2 Hardness of bicriteria approximation for MULTICUT

In this section we prove the part of Theorem 1.2 regarding the MULTICUT problem, namely, that the Unique Games Conjecture implies that it is NP-hard to approximate MULTICUT within a certain factor L. Our proof will actually show a stronger result — for every $\alpha \geq 7/8$ it is NP-hard to distinguish between whether there is a multicut of cost less than $n2^{d+1}$ (the YES instance) or whether every cutset that disconnects at least αk demand pairs has cost at least $n2^{d+1}L$ (the NO instance). This implies that it is NP-hard to obtain an (α, L)-bicriteria approximation for MULTICUT.

We start by describing a reduction from unique 2-prover game to MULTICUT (Section 2.1), and then proceed to analyze the YES instance (Section 2.2) and the NO instance (Sections 2.3 and 2.4). Finally, we discuss the gap that is created for a bicriteria approximation of MULTICUT (Section 2.5).

2.1 The reduction

Given a regular unique 2-prover game instance $G_Q = (Q, E_Q)$ with $n = |Q|/2$ and the corresponding edge weights w_{ij} and bijections $b_{ij} : [d] \to [d]$, we construct a MULTICUT instance $G = (V, E)$ with demand pairs, as follows. For every vertex (i.e., question) $q_i^p \in Q$, construct a d-dimensional hypercube C_j^p; the dimensions in this cube correspond to answers for the question q_j^p.[3] For each of the $2n$ hypercubes, we let the edges inside the hypercube have cost 1, and call them *hypercube edges.*

For each question edge $(q_i^1, q_j^2) \in E_Q$, we extend b_{ij} (in the obvious way) to a bijection from the vertices of C_i^1 to the vertices of C_j^2, and denote the resulting bijection by $b'_{ij} : \{0,1\}^d \to \{0,1\}^d$. Formally, for every $u \in \{0,1\}^d$ (vertex in C_i^1) and every $a \in [d]$, the a-th coordinate of $b'_{ij}(u)$ is given by $(b'_{ij}(u))_a = u_{b_{ij}^{-1}(a)}$. Then, we connect every vertex $u \in C_i^1$ to the corresponding vertex $b'_{ij}(u) \in C_j^2$ using an edge of cost $w_{ij}\Lambda$, where $\Lambda = n/\eta$ is a scaling factor. These edges are called *cross edges.*

Denote the resulting graph by $G = (V, E)$. Notice that V is simply the union of the vertex sets of the hypercubes C_i^p, for all $p \in [2]$ and $i \in [n]$, and that the edge set E contains two types of edges, hypercube edges and cross edges.

[3] This is a standard technique in PCP constructions for graph optimization problems. A hypercube can be interpreted as a "long code" [BGS98], and a dimension cut is the encoding of an answer in the 2-prover game.

To complete the reduction, it remains to define the demand pairs. For a vertex $u \in V$, the *antipode* of u in G, denoted $\overline{u}$, is defined to be the vertex antipodal to u in the hypercube C_i^p that contains u. The set D of demand pairs then contains every pair of antipodal vertices in G, and hence $k = |D| = n2^{d-1}$. Note that every vertex of G belongs to exactly one demand pair.

2.2 The YES instance

Lemma 2.1. *If there is a solution A for the unique 2-prover game G_Q such that the total weight of the satisfied questions is at least $1-\eta$, then the resulting* MULTICUT *instance G contains a multicut $M \subseteq E$ such that $c(M) \leq 2^{d+1}n$.*

Proof. Let A be such a solution for G_Q. Construct M by taking the following edges. For every question $q_i^p \in Q$ and the corresponding answer A_i^p (of prover p), take the dimension-A_i^p cut in cube C_i^p. In addition, for every edge $(q_i^1, q_j^2) \in E_Q$ that the solution A does not satisfy, take all the cross edges between the corresponding cubes C_i^1 and C_j^2.

We first claim that removing M from G disconnects all the demand pairs. To see this, we define a Boolean function $f : V \to \{0,1\}$ on the graph vertices. For every cube C_i^p, consider the dimension-A_i^p cut; it disconnects the cube into two connected components, one containing the all zeros vector $\mathbf{0}$ and one containing the all ones vector $\mathbf{1}$. For every $v \in C_i^p$, let $f(v) = 0$ if v is in the same side as $\mathbf{0}$, and $f(v) = 1$ otherwise. This is exactly the A_i^p-th bit in v, i.e., $f(v) = v_{A_i^p}$. Now consider any demand pair $(v, \overline{v})$, and note that $f(v) = 1 - f(\overline{v})$. We will show below that every edge $(u,v) \notin M$ satisfies the property $f(u) = f(v)$. This will clearly complete the proof of the claim.

Consider first a hypercube edge (u,v) in C_i^p that is not a dimension-A_i^p edge. Then $f(u) = u_{A_i^p} = v_{A_i^p} = f(v)$, by the definition of f. Next consider a cross edge $(u,v) \notin M$. Then this edge lies between cubes C_i^1 and C_j^2, such that the question edge (q_i^1, q_j^2) satisfied by the unique 2-prover game solution A. Therefore, $b_{ij}(A_i^1) = A_j^2$. Then, $f(u) = u_{A_i^1} = v_{b_{ij}(A_i^1)} = v_{A_j^2} = f(v)$.

Finally, we bound the cost of the solution. Let S be the set of question edges not satisfied by the solution A. The total cost of the multicut solution is thus

$$c(M) = 2n\, 2^{d-1} + 2^d \Lambda \sum_{(q_i^1, q_j^2) \in S} w_{ij} \leq 2^d n + 2^d \Lambda \eta = 2^{d+1} n.$$

□

2.3 Hypercube cuts, Boolean functions, influences, and juntas

We will analyze the NO instance shortly, but first we set up some notation and present a few technical lemmas regarding cuts in hypercubes. In particular, we present Theorem 2.2, which will have a crucial role in the sequel.

Recall that the dimensions of the hypercubes in the multicut instance correspond to answers to the 2-prover game. Therefore, we would like to determine which dimensions are the most significant participants in a cut on the cube, as follows. Let $C = (V_C, E_C)$ be a d-dimensional hypercube. It will be useful to represent cuts on the hypercube C as functions $f : V_C \to \mathbb{Z}$. Such a function f corresponds to a partition of V_C into sets $\{f^{-1}(r) : r \in f(V_C)\}$, which in turn corresponds to the cutset $\{(u,v) \in E_C \mid f(u) \neq f(v)\}$. Notice that f can be described as a function on d Boolean variables (corresponding to the dimensions of the hypercube), where the dimension $a \in [d]$ corresponds to the a-th variable. The *influence of a dimension (variable)* $a \in [d]$ on the function f, denoted I_a^f, is defined as the fraction of the dimension a-edges $(u,v) \in E_C$ for which $f(u) \neq f(v)$.

In other words, $I_a^f = \Pr_{u\in V_C}[f(u) \neq f(u \oplus 1_a)]$ where 1_a is a unit vector along dimension a. The *total influence* (also called average sensitivity) of f is $\sum_{a\in[d]} I_a^f$. We say that the function f is a *k-junta* if there exists a subset $J \subseteq [d]$, $|J| \leq k$, such that for every variable (dimension) $a \notin J$ and for every $u \in V_C$, we have $f(u) = f(u \oplus 1_a)$. In other words, f depends on at most k variables, and the remaining variables have zero influence. Two functions f and f' are said to be *ε-close* if $\Pr_{u\in V_C}[f(u) \neq f'(u)] \leq \varepsilon$.

An important special case is that of Boolean functions, i.e., $g : V_C \to \{0,1\}$, which corresponds to a bipartition of V_C. The *balance* of a Boolean function g is defined as $\min\left\{\frac{|g^{-1}(0)|}{|V_C|}, \frac{|g^{-1}(1)|}{|V_C|}\right\}$, i.e., the minimum of $\Pr_{u\in V_C}[g(u) = 0]$ and $\Pr_{u\in V_C}[g(u) = 1]$.

The next theorem, due to Friedgut [Fri98], asserts that every Boolean function of low total influence is close to a junta. We will later use it to determine a set of dimensions that are the most significant participants in a low-cost cutset.

Theorem 2.2 (Friedgut's Junta Theorem [Fri98]). *Let g be a Boolean function defined on a hypercube and fix $\varepsilon > 0$. Then g is ε-close to a Boolean function h defined on the same cube and depending on only $2^{O(T/\varepsilon)}$ variables, where $T = \sum_{a\in[d]} I_a^g$ is the total influence of g.*

2.4 The NO instance

Lemma 2.3. *There exists $L = \Omega(\min\{\eta^{-1}, \log\delta^{-1}\})$ such that if the* Multicut *instance G has a cutset of cost at most $2^{d+1}nL$ whose removal disconnects $\alpha \geq 7/8$ fraction of the demand pairs, then there is a solution A for the unique 2-prover game G_Q whose value is larger than δ.*

Proof. Let $L = c \cdot \min\{1/\eta, \log(1/\delta)\}$ for a constant $c > 0$ to be determined later, and let $M \subseteq E$ be a cutset of cost $c(M) \leq 2^{d+1}nL$ whose removal disconnects $\alpha \geq 7/8$ fraction of the demand pairs. Using M, we will construct for the unique 2-prover game G_Q a randomized solution A whose expected value is larger than δ, thereby proving the existence of a solution of value larger than δ. The randomized solution A, which is a strategy for the two provers, is defined as follows. Label each connected component of $G \setminus M$ as either 0 or 1 independently at random with equal probabilities, and define a Boolean function $f : V \to \{0,1\}$ by letting $f(v)$ be the label of the connected component of $v \in V$. Next, for each vertex (question) $q_i^p \in Q$ consider the restriction of f to the hypercube $C_i^p \subset V$, denoted $f_{|C_i^p}$, and apply to it Theorem 2.2 (Friedgut's Junta Theorem) with $\varepsilon = 1/60$, to obtain a subset of the variables (dimensions) $J_i^p \subseteq [d]$; the idea is that for many hypercubes we obtain $|J_i^p| \leq 2^{O(L/\varepsilon)}$. Finally, choose the answer A_i^p uniformly at random from J_i^p, independently of all other events.

We proceed to analyze the expected value of this randomized solution A. Recall that the value of a solution is equal to the probability that, for a question edge (q_i^1, q_j^2) chosen at random with probability proportional to its weight, we have $a_j^2 = b_{ij}(a_i^1)$. Notice that although q_i^1 and q_j^2 are correlated, each one is uniformly distributed because Q is regular. Without loss of generality, we assume removing M disconnects at least as many demand pairs inside the cubes $\{C_l^1\}_{l\in[n]}$ as inside the cubes $\{C_l^2\}_{l\in[n]}$. Now we claim that with a constant probability over the choice of a question edge, the cut M has a low cost over edges incident on the two corresponding hypercubes, and disconnects many demand pairs in these two hypercubes. In other words, the quality of the cut locally is nearly as good as the quality of the cut globally. In particular, we upper bound the probability of the following four "bad" events (for a random question edge (q_i^1, q_j^2)):

$\mathcal{E}_1$ = fewer than 1/8-fraction of the vertices $v \in C_i^1$ satisfy $f(v) \neq f(\overline{v})$.

$\mathcal{E}_2$ = M contains more than $2^{d+5}L$ hypercube edges in C_i^1.

$\mathcal{E}_3$ $= M$ contains more than $2^{d+5}L$ hypercube edges in C_j^2.

$\mathcal{E}_4$ $= M$ contains more than $2^{d+5}\eta L$ cross edges between C_i^1 and C_j^2.

First, by our assumption above, removing M disconnects at least $\alpha \geq 7/8$ fraction of the demand pairs inside the cubes $\{C_l^1\}_{l\in[n]}$. Thus, the expected fraction of demand pairs $(v,\overline{v})$ in C_i^1 that are not disconnected in $G \setminus M$ (and thus clearly $f(v) = f(\overline{v})$) is at most 1/8. In addition, the expected fraction of demand pairs $(v,\overline{v})$ in C_i^1 that are disconnected in $G \setminus M$ and satisfy $f(v) = f(\overline{v})$, is at most 1/2, because different connected components of $G \setminus M$ are labeled independently. Thus, the expected fraction of vertices $v \in C_i^1$ for which $f(v) = f(\overline{v})$ is at most 5/8, and by Markov's inequality, $\Pr[\mathcal{E}_1] \leq 5/7$. Next, the cutset M contains at most $2^{d+1}nL$ hypercube edges, thus the expected number of edges in $C_i^1 \cup C_j^2$ that are contained in M is at most $2^{d+1}L$, and by Markov's inequality $\Pr[\mathcal{E}_2 \cup \mathcal{E}_3] \leq 1/16$. Finally, $\Pr[\mathcal{E}_4] \leq 1/16$, as otherwise the total cost of M along the cross edges corresponding to this event is more than $1/16 \cdot (2^{d+5}\eta L) \cdot \Lambda = 2^{d+1}nL \geq c(M)$. Taking a union bound, we upper bound the probability that any of the bad events occurs by

$$\Pr[\mathcal{E}_1 \cup \mathcal{E}_2 \cup \mathcal{E}_3 \cup \mathcal{E}_4] \leq \frac{5}{7} + \frac{2}{16} < \frac{6}{7}.$$

In order to lower bound the expected value of the randomized solution A, we would like to show that if none of the four bad events above happens, then the two sets of dimensions J_i^1 and J_j^2 obtained using Friedgut's Junta Theorem are relatively small, and further they are in "weak agreement", and these two properties will immediately imply that the randomized solution A satisfies $\Pr[b_{ij}(A_i^1) = A_j^2] > \delta$. Observe that if $(u,v) \in E$ and $f(u) \neq f(v)$, then $(u,v) \in M$. If the event $\mathcal{E}_2$ does not occur, then the total influence of $f_{|C_i^1}$ is at most $64L$, and thus $|J_i^1| \leq 2^{O(L/\varepsilon)}$. Similarly, if the event $\mathcal{E}_3$ does not occur, then the total influence of $f_{|C_j^2}$ is at most $64L$, and thus $|J_j^2| \leq 2^{O(L/\varepsilon)}$. In addition, if the event $\mathcal{E}_1$ does not occur, then the balance of $f_{|C_i^1}$ is at least 1/16.

We now claim that if none of the above bad events happens then there is $a \in J_i^1$ for which $b_{ij}(a) \in J_j^2$. Indeed, assume towards contradiction that $J_i^1 \cap b_{ij}^{-1}(J_j^2) = \emptyset$. Then by construction there is a Boolean function $g_i^1 : C_i^1 \to \{0,1\}$ that is ε-close to $f_{|C_i^1}$ and depends on only variables in J_i^1. Clearly, the balance of g_i^1 is close to that of $f_{|C_i^1}$, namely, at least $1/16 - \varepsilon$. Similarly, there is a Boolean function $g_j^2 : C_j^2 \to \{0,1\}$ that is ε-close to $f_{|C_j^2}$ and depends on only variables in J_j^2. We can relate these two functions via $b'_{ij} : C_1^i \to C_j^2$, namely by considering $h : C_i^1 \to \{0,1\}$ given by $h(v) = g_j^2(b'_{ij}(v))$.

Notice that h is ε-close to $f_{|C_j^2} \circ b'_{ij}$, and that it depends only on variables in $b_{ij}^{-1}(J_j^2)$. Therefore g_i^1 and h depend on disjoint sets of variables. It follows that $\Pr_{v\in C_i^1}[g_i^1(v) \neq h(v)] \geq 1/16 - \varepsilon$, because if we condition on the value of the variables in $b_{ij}^{-1}(J_j^2)$ we get that $h(v)$ is determined, but this does not affect the distribution of $g_i^1(v)$, which still attains each value (0 or 1) with probability at least $1/16 - \varepsilon$. Consequently, g_i^1 and $h = g_j^2 \circ b'_{ij}$ are not say $(1/17 - \varepsilon)$-close.

On the other hand, the event $\mathcal{E}_4$ not occurring implies that at most $32\eta L$ vertices $v \in C_i^1$ satisfy $f(v) \neq f(b'_{ij}(v))$. In other words, $f_{|C_i^1}$ is $(32\eta L)$-close to $f_{|C_j^2} \circ b'_{ij}$. The former is ε-close to g_i^1 while the latter is ε-close to $g_j^2 \circ b'_{ij}$ (because b'_{ij} is a bijection on the variables), and by the triangle inequality we get that g_i^1 and $h = g_j^2 \circ b'_{ij}$ are $2\varepsilon + 32\eta L$ close. If $c > 0$ is sufficiently small, $2\varepsilon + 32\eta L < 1/17 - \varepsilon$, which yields a contradiction and completes the proof of the claim.

Using the above claim we get that for a random question edge,

$$\Pr[A_j^2 = b_{ij}(A_i^1)] \quad \geq \quad \Pr[A_i^1 \in J_i^1 \cap b_{ij}^{-1}(J_j^2), A_j^2 = b_{ij}(A_i^1)]$$

$$
\begin{aligned}
&\geq \frac{1}{7} \cdot 2^{-O(L/\varepsilon)} \cdot 2^{-O(L/\varepsilon)} \\
&= 2^{-O(L)}.
\end{aligned}
$$

We conclude that the expected value of the randomized solution A is $\Pr[A_j^2 = b_{ij}(A_i^1)] \geq 2^{-O(L)} > \delta$, where the last inequality holds if $c > 0$ is sufficiently small, and this completes the proof of Lemma 2.3. $\square$

2.5 Putting it all together

The above reduction from unique 2-prover game to MULTICUT produces a gap of $L(n) = \Omega(\min\{\frac{1}{\eta(n)}, \log \frac{1}{\delta(n)}\})$. We assumed $d(\eta, \delta) \leq O(\log n)$, and thus the resulting MULTICUT instance G has size $N = (n2^d)^{O(1)} = n^{\Theta(1)}$. It follows that in terms of the MULTICUT instance size N, the gap is

$$L(N) = \Omega(\min\{\tfrac{1}{\eta(N^{\Theta(1)})}, \log \tfrac{1}{\delta(N^{\Theta(1)})}\}).$$

This completes the proof of the part of Theorem 1.2 regarding the MULTICUT problem, namely, that the Unique Games Conjecture implies that it is NP-hard to approximate MULTICUT within the above factor $L(N)$. In fact, the above proof shows that it is NP-hard to obtain even a $(7/8, L(N))$-bicriteria approximation.

Note that the number of demand pairs is $k = n2^{d-1} = n^{\Theta(1)}$, and thus the hardness of approximation factor is similar when expressed in terms of k as well. Note also that all edge weights in the MULTICUT instance constructed above are bounded by a polynomial in the size of the graph. Therefore, via a standard reduction, a similar hardness result holds for the unweighted MULTICUT problem as well.

3 Hardness of approximating SPARSEST-CUT

In this section we prove the part of Theorem 1.2 regarding the Sparsest-Cut problem. The proof follows immediately from the next lemma in conjunction with the hardness of bicriteria approximation of MULTICUT (from the Section 2).

Lemma 3.1. *Let $0 < \alpha < 1$ be a constant. If there exists a polynomial-time algorithm for* SPARSEST-CUT *that produces a cut whose value is within factor $\rho \geq 1$ of the minimum, then there is a polynomial time algorithm that computes an $(\alpha, \frac{\rho}{1-\alpha})$-bicriteria approximation for* MULTICUT.

Proof. Fix $0 < \alpha < 1$, and suppose $\mathcal{A}$ is a polynomial-time algorithm for SPARSEST-CUT that produces a cut whose value is within factor $\rho \geq 1$ of the minimum. Now suppose we are given an input graph $G = (V, E)$ and k demand pairs $D = \{\{s_1, t_1\}, \ldots, \{s_k, t_k\}\}$. We may assume without loss of generality that every s_i is connected (in G) to its corresponding t_i.

We now describe the bicriteria approximation algorithm for MULTICUT. It is an iterative algorithm where $M = \emptyset$ at the beginning and the following is executed at each iteration: we apply algorithm $\mathcal{A}$ to G with D as the set of demand pairs, letting M' be the cutset produced by A and D' be the demand pairs in D that are disconnected by M', and then set $M \leftarrow M \cup M'$ and $D \leftarrow D \setminus D'$. Note that D changes throughout the iterations, but G does not. The iterations proceeds as long as $|D| \geq (1 - \alpha)k$, and eventually the final M is output.

This algorithm clearly runs in polynomial time, and there are at most αk iterations. For the sake of analysis, fix an optimal multicut $M^* \subseteq E$, i.e., a cutset of G whose removal disconnects all the demand pairs and has the least cost. We first claim that in every iteration, there is a cut whose value (in the Sparsest-Cut sense) is at most $c(M^*)/((1-\alpha)k)$; this follows since $|D| \geq (1-\alpha)k$ and M^* disconnects all the demand pairs in D. By the approximation guarantees of $\mathcal{A}$, we have $c(M') \leq \rho c(M^*)/((1-\alpha)k)$ at every iteration. Summing over all the iterations, we can upper bound the cost of the output cut by

$$c(M) \leq \rho \cdot \frac{c(M^*)}{(1-\alpha)k} \cdot k = \frac{\rho}{1-\alpha} c(M^*).$$

Clearly, the output cut disconnects at least αk demand pairs in G, and this concludes the proof of the lemma. ☐

4 Hardness of approximating Min-2CNF≡ Deletion

In this section, we modify the reduction in Section 2.1 to obtain a hardness of approximation for Min-2CNF≡ Deletion. In particular, we reduce the Multicut instance obtained in Section 2.1 to Min-2CNF≡ Deletion, such that a solution to the latter gives a Multicut of the same cost in the former.

The Min-2CNF≡ Deletion instance contains $2^{d-1}n$ variables, one for each demand pair $\{u, \overline{u}\}$. In particular, for every demand pair $\{u, \overline{u}\} \in D$, we associate the literal x_u with u and the literal $x_{\overline{u}} = \neg x_u$ with $\overline{u}$. For every edge $e = (u, v)$ in the graph G there is a clause $(x_u \equiv x_v)$ whose weight is equal to the edge-weight w_e.

The following lemma is immediate from the construction and implies an analog of Lemma 2.3 for Min-2CNF≡ Deletion.

Lemma 4.1. *Given an assignment S of cost W to the above instance of* Min-2CNF≡ Deletion*, we can construct a solution of cost W to the* Multicut *instance G.*

Proof. Let M be the set of edges (u, v) for which $S(x_u) \neq S(x_v)$. Then M corresponds to the clauses that are not satisfied by S and has weight W. The lemma follows from observing that M is indeed a multicut — S is constant over connected components in $G \setminus M$, and for every demand pair $(u, \overline{u})$, we have $S(x_u) \neq S(x_{\overline{u}})$. ☐

We now give an analog of Lemma 2.1.

Lemma 4.2. *If there is a solution A for the unique 2-prover game G_Q such that the total weight of the satisfied questions is at least $1-\eta$, then there exists an assignment S for the above* Min-2CNF≡ Deletion *instance such that $c(S) \leq 2^{d+1}n$.*

Proof. Given the solution A for G_Q, we construct an assignment S as follows. For every question q_i^p and for every vertex u in the corresponding hypercube C_i^p, define $S(x_u)$ to be the A_i^p-th bit of u, i.e., $S(x_u) = u_{A_i^p}$. Note that this is a valid assignment, i.e., $S(x_u) = 1 - S(x_{\overline{u}})$ for all vertices u, as $u_{A_i^p} = 1 - \overline{u}_{A_i^p}$.

We bound the cost of the solution by first analyzing the clauses corresponding to hypercube edges in the corresponding Multicut instance. Consider unsatisfied clauses containing both variables in the same hypercube C_i^p, and note that the hypercube edges corresponding to these clauses form a dimension-A_i^p cut in the cube C_i^p. Therefore, the total weight of these clauses is at most $(2^{d-1})(2n) = 2^d n$.

Finally, consider an unsatisfied clause $(x_u \equiv x_v)$ corresponding to vertices in different hypercubes C_i^1 and C_j^2. Then $S(x_u) \neq S(x_v)$ implies that $u_{A_i^1} = v_{b_{ij}(A_i^1)} \neq v_{A_j^2}$, or, $b_{ij}(A_i^1) \neq A_j^2$. There are at most 2^d such clauses for each question pair not satisfied by the solution A. Therefore, the total weight of such clauses is at most $2^d \eta \Lambda = 2^d n$.

The lemma follows from adding the total weight of these two types of edges. □

Lemmas 4.1 and 4.2 along with Lemma 2.3 imply the part of Theorem 1.2 regarding Min-2CNF≡ Deletion.

5 Concluding remarks

Several important questions are left open. First, one would like to eliminate the dependence on the Unique Games Conjecture, and obtain a "standard" hardness of approximation result. Yet another challenge is to improve the hardness factor. For Multicut, the $\Omega(\log k)$ integrality ratio lower bound of [ACMM05] suggests that the inapproximability bound may be improved. In particular, $(\log k)^c$ hardness for a constant $c > 1/2$ will separate the approximability of Multicut from that of Sparsest-Cut (in light of the recent approximation due to [ALN05]). The main bottleneck to improving the hardness factor lies in Friedgut's Junta Theorem (and similarly in the result of [KKL88] that we used in the conference version). These bounds are tight in general, as shown by the tribes function [BL90] (see also [Fri98, Section 2]).

A third challenge is to obtain hardness of approximation results for the uniform-demand case of the Sparsest-Cut problem or for the Balanced-Cut problem. Our results do not apply to this special but important case; in particular, if a 2-prover system has a low-cost balanced cut, then the corresponding graph on hypercubes would have a low-cost balanced cut regardless of the value of the 2-prover game. Very recently, Devanur et al. [DKSV06] have shown a lower bound of $\Omega(\log \log n)$ on the integrality ratio of the natural SDP relaxation for this problem. Alternatively, of course, one might improve the approximation algorithms for any of these problems.

Acknowledgements We thank the anonymous referees for useful comments, and in particular for suggesting a simplification of our original proof of Lemma 3.1.

References

[ACMM05] A. Agarwal, M. Charikar, K. Makarychev, and Y. Makarychev. $O(\sqrt{\log n})$ approximation algorithms for Min UnCut, Min 2CNF Deletion, and directed cut problems. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 573–581, 2005. 1.1, 5

[ALM+98] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM*, 45(3):501–555, 1998. 1.2

[ALN05] S. Arora, J. R. Lee, and A. Naor. Euclidean distortion and the sparsest cut. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 553–562, 2005. 1.1, 5

[AR98] Y. Aumann and Y. Rabani. An $O(\log k)$ approximate min-cut max-flow theorem and approximation algorithm. *SIAM Journal on Computing*, 27(1):291–301, 1998. 1.1

[ARV04] S. Arora, S. Rao, and U. Vazirani. Expander flows, geometric embeddings, and graph partitionings. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*, pages 222–231, 2004. 1.1

[AS98] S. Arora and S. Safra. Probabilistic checking of proofs: A new characterization of NP. *Journal of the ACM*, 45(1):70–122, 1998. 1.2

[BBC04] N. Bansal, A. Blum, and S. Chawla. Correlation Clustering. *Machine Learning, Special Issue on Clustering*, 56(1-3):89–113, 2004. 1

[BGS98] M. Bellare, O. Goldreich, and M. Sudan. Free bits, PCP's and non-approximability – towards tight results. *SIAM Journal on Computing*, 27(3):804–915, 1998. 3

[BL90] M. Ben-Or and N. Linial. Collective coin flipping. In *Randomness and Computation*, pages 91–115, 1990. 5

[CGR05] S. Chawla, A. Gupta, and H. Räcke. Improved approximations to sparsest cut. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 102–111, 2005. 1.1

[CGW03] M. Charikar, V. Guruswami, and A. Wirth. Clustering with qualitative information. In *Proceedings of the 44th IEEE Symposium on Foundations of Computer Science*, pages 524–533, 2003. 1

[CKK+05] S. Chawla, R. Krauthgamer, R. Kumar, Y. Rabani, and D. Sivakumar. On the hardness of approximating multicut and sparsest-cut. In *Proceedings of the 20th Annual IEEE Conference on Computational Complexity*, pages 144–153, June 2005. 1.3

[CMM06] M. Charikar, K. Makarychev, and Y. Makarychev. Near-optimal algorithms for Unique Games. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, 2006. To appear. 1.2, 1.3

[DI03] E. Demaine and N. Immorlica. Correlation clustering with partial information. In *Proceedings of the 6th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX)*, pages 1–13, 2003. 1

[DJP+94] E. Dahlhaus, D. S. Johnson, C. H. Papadimitriou, P. D. Seymour, and M. Yannakakis. The complexity of multiterminal cuts. *SIAM Journal on Computing*, 23(4):864–894, 1994. 1, 1.1

[DKSV06] N. Devanur, S. Khot, R. Saket, and N. Vishnoi. Integrality gaps for sparsest cut and minimum linear arrangement problems. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, 2006. To appear. 5

[EF03] D. Emanuel and A. Fiat. Correlation clustering—minimizing disagreements on arbitrary weighted graphs. In *Proceedings of 11th Annual European Symposium on Algorithms*, pages 208–220, 2003. 1

[FR04] U. Feige and D. Reichman. On systems of linear equations with two variables per equation. In *Proceedings of the 7th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX)*, pages 117–127, 2004. 1.2

[Fri98] E. Friedgut. Boolean functions with low average sensitivity depend on few coordinates. *Combinatorica*, 18(1):27–35, 1998. 1.3, 2.3, 2.2, 5

[GT06] A. Gupta and K. Talwar. Approximating unique games. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 99–106, 2006. 1.2

[GVY96] N. Garg, V. V. Vazirani, and M. Yannakakis. Approximate max-flow min-(multi)cut theorems and their applications. *SIAM Journal on Computing*, 25(2):235–251, 1996. 1.1

[Hås01] J. Håstad. Some optimal inapproximability results. *Journal of the ACM*, 48(4):798–859, 2001. 1.1

[KARR90] P. Klein, A. Agrawal, R. Ravi, and S. Rao. Approximation through multicommodity flow. In *Proceedings of the 31st IEEE Symposium on Foundations of Computer Science*, pages 726–737, 1990. 1.1

[Kho02] S. Khot. On the power of unique 2-prover 1-round games. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, pages 767–775, 2002. 1, 1.1, 1.1, 1.2

[KKL88] J. Kahn, G. Kalai, and N. Linial. The influence of variables on boolean functions. In *Proceedings of the 29th IEEE Symposium on Foundations of Computer Science*, pages 68–80, 1988. 1.3, 5

[KKMO04] S. Khot, G. Kindler, E. Mossel, and R. O'Donnell. Optimal inapproximability results for MAX-CUT and other 2-variable CSPs. In *Proceedings of the 45th IEEE Symposium on Foundations of Computer Science*, pages 146–154, 2004. See also ECCC Report TR05-101, 2005. 1.2

[KR03] S. Khot and O. Regev. Vertex cover might be hard to approximate to within 2-ϵ. In *Proceedings of the 18th Annual IEEE Conference on Computational Complexity*, pages 379–386, 2003. 1.2, 1.4

[KR06] R. Krauthgamer and Y. Rabani. Improved lower bounds for embeddings into l_1. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithm*, pages 1010–1017. ACM, 2006. 1.1

[KV05] S. Khot and N. K. Vishnoi. The unique games conjecture, integrality gap for cut problems and the embeddability of negative type metrics into ℓ_1. In *Proceedings of the 46th IEEE Symposium on Foundations of Computer Science*, pages 53–62, 2005. 1.1, 1.2, 1.3, 1.3

[LLR95] N. Linial, E. London, and Y. Rabinovich. The geometry of graphs and some of its algorithmic applications. *Combinatorica*, 15(2):215–245, 1995. 1.1

[LR88] F. T. Leighton and S. Rao. An approximate max-flow min-cut theorem for uniform multicommodity flow problems with applications to approximation algorithms. In *Proceedings of the 29th IEEE Symposium on Foundations of Computer Science*, pages 422–431, 1988. 1.1

[MOO05] E. Mossel, R. O'Donnell, and K. Oleszkiewicz. Noise stability of functions with low influences: Invariance and optimality. In *Proceedings of the 46th IEEE Symposium on Foundations of Computer Science*, pages 21–30, 2005. 1.2

[Raz98] R. Raz. A parallel repetition theorem. *SIAM Journal on Computing*, 27(3):763–803, 1998. 1.2

[Tre05] L. Trevisan. Approximation algorithms for unique games. In *Proceedings of the 46th IEEE Symposium on Foundations of Computer Science*, pages 197–205, 2005. See also comment for ECCC TR05-34. 1.2

[Vaz01] V. V. Vazirani. *Approximation algorithms.* Springer-Verlag, Berlin, 2001. 1.1

A Regularity of the Unique Games instance

Proof of Lemma 1.5. Given a unique 2-prover game Q, we describe how to convert it to a regular game while (nearly) preserving its completeness and soundness. Let the maximum weight of any edge in Q be $w_{\max} = \max_e w_e$. Let $r = \max\{\frac{1}{\eta}, \frac{1}{\delta}\}$. First we remove all edges of weight less than $\frac{1}{2n^2 r} w_{\max}$ from the graph, and renormalize the weights of the remaining edges so that they still sum to 1. Note that the optimal solution to the game has value at least $w_{\max}$, and this edge-removal step reduces the value of any solution by at most $\frac{1}{2r} w_{\max}$. So any solution of value $1-\eta$ in the original game now has value at least $1 - \eta - \frac{1}{2r} > 1 - \frac{3}{2}\eta$, and any solution of value at most δ in the original game now has value at most $\delta(1 + \frac{1}{2r}) < \frac{3}{2}\delta$.

Next, let $w_{\min}$ be the minimum weight of any edge. We round down weights of all edges to the nearest multiple of $t = \frac{1}{2r} w_{\min}$, and renormalize the weights so that they sum to 1. As before, we only remove a $\frac{1}{2r}$ fraction of the total weight and renormalize by a factor of at most $1 + \frac{1}{2r}$. Therefore, this changes the soundness and completeness parameters by at most an additive $\frac{1}{2r}$, as before. Over all, the parameters change by a factor of at most 2.

Now we convert the pre-processed game Q to a regular graph Q' as follows. For each prover $p \in \{1, 2\}$ and question q_i^p, form $W(p,i)/t$ vertices $q_i^p(1), \cdots, q_i^p(W(p,i)/t)$, where $W(p,i)$ is the total weight of all the edges incident on q_i^p. For every pair of vertices (q_i^1, q_j^2), connected by an edge e in Q, we form an edge between $q_i^1(x)$ and $q_j^2(y)$, for all possible values of x and y, with weight $w_e \frac{t}{W(1,i)} \frac{t}{W(2,j)}$. The bijection on the edge $(q_i^1(x), q_j^2(y))$ is set to be the same as the original bijection b_{ij} on edge (q_i^1, q_j^2).

Note that the total weight of all the edges remains the same as before. Each new vertex $q_i^1(x)$ has total weight $\sum_e w_e \frac{t}{W(1,i)} \frac{t}{W(2,j)} \frac{W(2,j)}{t} = t$, where the sum is over all edges e incident on q_i^1. The same holds for any new vertex $q_j^2(y)$. Therefore, the graph is regular. Furthermore, the number of vertices increases by a factor of at most $4n^2r^2$.

It only remains to show that the soundness and completeness parameters are preserved. To see this, note that any solution on the original graph Q can be transformed to a solution of the same value on Q', by picking the same answer for every vertex $q_i^p(x)$ in Q' as the answer picked for q_i^p in Q. Likewise, consider a solution in Q'. Note that for a fixed p and i, all questions $q_i^p(x)$ are connected to identical sets of vertices with identical bijections. Thus, we can replace answers to all these questions by the answer to the question $q_i^p(x)$ that has the most satisfied edges incident on it, thereby increasing the value of the solution. Now, the solution in Q that picks the same answer for q_i^p as the answer for $q_i^p(x)$ in Q' has the same weight as the new solution in Q' which is at least as much as the value of the original solution in Q'.

Thus for every solution in Q (after pre-processing), there is a solution of the same or larger weight in Q' and vice versa. This proves that the two games have exactly the same soundness and completeness parameters. ∎