On the Hardness of Approximating Multicut and Sparsest-Cut

Shuchi Chawla* Robert Krauthgamer† Ravi Kumar† Yuval Rabani‡ D. Sivakumar†

November 18, 2004

Abstract

We show that the MULTICUT, SPARSEST-CUT and MIN-2CNF≡ DELETION problems are hard to approximate, assuming the Unique Games Conjecture of Khot [Kho02]. In particular, we obtain an arbitrarily large constant factor hardness for these problems, and show that a quantitatively stronger version of the conjecture implies a hardness factor of $\Omega(\log\log n)$.

1 Introduction

In the MULTICUT problem the input is an undirected graph $G = (V, E)$ on $n = |V|$ vertices and k pairs of vertices $\{s_i, t_i\}_{i=1}^k$, called *demand pairs*, and the goal is to find a smallest 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 multicut M 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 a strong version of the Unique Games Conjecture of Khot [Kho02] is true, then MULTICUT is NP-hard to approximate to within a factor of $\Omega(\log\log n)$. Under the original version of this conjecture, our reduction shows that for every constant $L > 0$, it is NP-hard to approximate MULTICUT to within factor L.

Our methods also yield similar bounds for SPARSEST-CUT and for 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 $(V, E \setminus M)$. Since SPARSEST-CUT is not 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 $x = y$, where x and y are literals, and the goal is to find an assignment to the variables minimizing the total weight of unsatisfied clauses. Our results also extend to the CORRELATION CLUSTERING problem [BBC04, CGW03, DI03, EF03] of minimizing disagreements in a weighted graph, because this problem is known to be equivalent to the MULTICUT problem on weighted graphs.

*Computer Science Department, Carnegie Mellon University, Pittsburgh, PA 15213, USA. Email: shuchi@cs.cmu.edu.
This work is supported in part by an IBM PhD Fellowship, and by NSF grants CCR-0122581 and IIS-0121678, and part of it was done while the author was visiting IBM Almaden Research Center.
†IBM Almaden Research Center, 650 Harry Road, San Jose, CA 95120, USA. Email: {robi,ravi,siva}@almaden.ibm.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

1.1 Some known results on MULTICUT, SPARSEST-CUT, and MIN-2CNF≡ DELETION

MULTICUT and SPARSEST-CUT are fundamental combinatorial problems, with connections to multicommodity flow, expansion, and metric embeddings. Both problems can be approximated to within an $O(\log k)$ factor through linear programming relaxations [LR88, GVY96, AR98, LLR95]. These bounds match the lower bounds on the integrality gaps up to constant factors [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], who give an approximation-preserving reduction from this problem to MULTICUT. Recently, starting with the ground-breaking $O(\sqrt{\log n})$-approximation for the uniform demands case [ARV04], improved approximation algorithms have been developed for the SPARSEST-CUT problem using a semidefinite programming relaxation [ARV04, CGR05, ALN04]. The best approximation factor currently known for general demands is $O(\sqrt{\log k}\log\log k)$ [ALN04]. The obvious modification of the semidefinite program used for SPARSEST-CUT to solve MULTICUT was recently shown to have an integrality ratio of $\Omega(\log k)$ [ACMM04], which matches 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 much smaller than 2. Also MIN-2CNF≡ DELETION problem is known to be APX-hard, as follows, e.g., from [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 noted in [Kho02], and are weaker than our results in several respects. First, our quantitative bounds are better, and thus if a stronger, yet almost as plausible, version of this conjecture is true, then our lower bound on the approximation factor improves to $L = \Omega(\log\log n)$, compared with the roughly $\Omega((\log\log n)^{1/4})$ hardness that follows 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.

For SPARSEST-CUT, no hardness of approximation result was previously known. Independent of our work, Khot and Vishnoi [KV04] 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 $(\log\log n)^c$, for some constant $c > 0$, assuming a stronger quantitative version of the conjecture. Additionally, they prove an integrality ratio lower bound of $\Omega((\log\log\log n)^c)$, for some fixed $c > 0$, for the semidefinite program relaxations used in the recent SPARSEST-CUT results.

1.2 The Unique Games Conjecture

Unique 2*-prover game* is the following problem. The input is a bipartite graph $G = (Q, E_Q)$, 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 $w(q_i^1, q_j^2)$. These edges will be called *question edges*, to distinguish them from edges in the MULTICUT instance. Each question is associated with a set of d distinct answers, denoted by $[d] = \{1, \ldots, d\}$. The input also contains, for every edge $(q_i^1, q_j^2) \in E_Q$, a bijection $b_{ij} : [d] \to [d]$, which maps every answer of question q_i^1 to a distinct answer for 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 an edge $(q_i^1, q_j^2) \in E_Q$ if the answers A_i^1 and A_j^2 agree, 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 $\max\{\eta, \delta\} \le 1/(\log n)^{\Omega(1)}$ and $d = d(\eta, \delta) \le O(\log n)$. We denote the size of an input instance by N. Notice that $N = (nd)^{\Theta(1)}$, and is thus polynomial in n as long as $d \le O(n)$, and in particular for fixed d.

Plausibility of the conjecture and its stronger version. The Unique Games Conjecture has been used to show optimal inapproximability results for VERTEX COVER [KR03] and MAX-CUT [KKMO04]. Proving the conjecture using current techniques appears 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, then necessarily $d \ge \max\{1/\eta^{1/10}, 1/\delta\}$. This follows from a semidefinite programming algorithm presented in [Kho02]. Our $\Omega(\log\log n)$ hardness result (see Corollary 1.4 below) requires the existence of a constant $c > 0$, such that $\max\{\eta, \delta\} \le 1/(\log n)^c$ and $d \le O(\log n)$, which is not excluded by the above. Feige and Reichman [FR04] recently 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.

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 $d = d(\eta, \delta) \le O(\log n)$, 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 exists $L(n) = \Omega\left(\log \frac{1}{\eta(n^{\Omega(1)}) + \delta(n^{\Omega(1)})}\right)$ such that it is NP-hard to approximate* MULTICUT, SPARSEST-CUT, *and* MIN-2CNF≡ DELETION *to within factor $L(n)$.*

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. *The stronger version of the Unique Games Conjecture in which $\max\{\eta, \delta\} \le 1/(\log n)^{\Omega(1)}$, and $d = d(\eta, \delta) \le O(\log n)$, implies that for some fixed $c > 0$, it is NP-hard to approximate* MULTICUT, SPARSEST-CUT, *and* MIN-2CNF≡ DELETION *to within factor $c \log\log n$.*

For SPARSEST-CUT our hardness results hold only for the search version (in which the algorithm needs to produce a cutset and not only its value), since our proof employs a Cook reduction.

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 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].

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 throughout a *cutset* of the graph. A cutset whose removal disconnects all the demand pairs is called 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 an α fraction of the demands and has cost at most β times the weight of the optimum multicut. In other words, if M^* is the 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 $C = (V_C, E_C)$ which has the vertex set $V_C = \{0,1\}^d$, and an edge $(u, v) \in E_C$ for every two vertices $u, v \in \{0,1\}^d$ which 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 a hypercube is called the *dimension-a cut* in the hypercube. The *antipodal* of a vertex u is the (unique) vertex $\overline{u}$ all of whose coordinates are different from those of u, i.e., the vertex $u \oplus \underline{1}$ where $\underline{1}$ is the vector with 1 in every coordinate. Notice that v is the antipodal of u if and only if u is the antipodal of v, and that every 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 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 unique 2-prover game instance $G_Q = (Q, E_Q)$ with $n = |Q|/2$ and the corresponding edge weights $w(e)$ 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. We let the edges insides these $2n$ cubes 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} to a bijection from the vertices of C_i^1 (subsets of the answers for q_i^1) to the vertices of C_j^2 (subsets of the answers for q_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 $v \in C_i^1$ to the corresponding vertex $b'_{ij}(u) \in C_j^2$ using an edge of cost $w_{ij}\Lambda$, where $\Lambda = \frac{n}{\eta}$ 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.

To complete the reduction, it remains to define the demand pairs. For a vertex $u \in V$, the *antipodal* of u in G, denoted $\overline{u}$, is defined to be the antipodal vertex of 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 belong 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 there exists a multicut $M \subseteq E$ for the* MULTICUT *instance G such that $c(M) \le 2^{d+1}n$.*

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 $\underline{0}$ and one containing the all ones vector $\underline{1}$. For every $v \in C_i^p$, let $f(v) = 0$ if v is in the same side as $\underline{0}$, 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 clearly proves 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 $c(M) = 2n\, 2^{d-1} + 2^d \Lambda \sum_{(Q_i^1, Q_j^2) \in S} w_{ij} \le 2^d n + 2^d \frac{n}{\eta} \eta = 2^{d+1} n$. ◻

2.3 Hypercube cuts and influences

We will analyze the NO instance shortly, but first we set up some notation and present two crucial technical lemmas regarding hypercubes. Let $H = (V_H, E_H)$ be a d-dimensional hypercube. For a function $f : V_H \to \mathbb{R}$, *the influence of dimension $a \in [d]$ (a.k.a. the influence of the a-th variable) on the function*, denoted I_a^f, is defined to be the fraction of dimension-a edges $(u, v) \in E_H$ for which $f(u) \neq f(v)$. For a cutset $M \subseteq E_H$, *the influence of dimension $a \in [d]$ on the cutset*, denoted I_a^M, is defined as the fraction of dimension-a edges that belong to M. Observe that $|M| = 2^{d-1} \sum_{a\in[d]} I_a^M$.

Proposition 2.2. *Let $M \subseteq E_H$ be a cutset in a hypercube $H = (V_H, E_H)$. Define $g : V_H \to \mathbb{Z}$ by labeling the connected components of $H \setminus M$ by distinct integers, and letting $g(v)$ for $v \in V_H$ be the label of the connected component containing v. Then $I_a^M \geq I_a^g$.*

Proof. Observe that the cutset M must contain every edge $(u, v) \in E_H$ for which $g(u) \neq g(v)$. □

The first lemma shows that if a cutset M has few edges but its removal disconnects a large fraction of the antipodal pairs in the hypercube H, then there must be a dimension $a \in [d]$ with large influence.

Lemma 2.3. *Let M be a cutset in a d-dimensional hypercube H, and suppose that removing M disconnects at least a β fraction of the antipodal pairs in H. For every $x > 0$, if $\sum_a I_a^M \leq \beta x$ then $\max_a I_a^M \geq 2^{-6x}/27$.*

We will make use of the following lemma, due to Kahn, Kalai and Linial [KKL98] (see also [Ste00, Section 1.5]). We note the proof is based on Fourier analysis of Boolean functions, and that our statement below follows from the proofs therein.

Lemma 2.4 (Kahn, Kalai, and Linial [KKL98]). *Let f be a Boolean function defined on a hypercube, and suppose the fraction of inputs x for which $f(x) = 1$ is $p \leq 1/2$. Then for all $\alpha > 0$,*

$$\frac{1}{\alpha}\sum_i I_i^f + \sum_i (I_i^f)^{4/3} \geq 2p\frac{\log\alpha}{\alpha}.$$

Proof of Lemma 2.3. We first convert the cutset M into a two-sided cut. Observe that each connected component of $H \setminus M$ must have size at most $2^d - \beta 2^{d-1} = (1-\beta/2)|V_H|$. If there is a component of size larger than $|V_H|/2$, we combine the rest of the components into a single component. Otherwise, we split the set of components into two parts such that the total size of the components in each part is at most $\frac{2}{3}|V_H|$. Call the resulting cutset M'. Note that $M' \subseteq M$ and thus, for every $a \in [d]$, the influence of every dimension in M' is no larger than its counterpart in M, i.e., $I_a^{M'} \leq I_a^M$. Hence, $\sum_a I_a^{M'} \leq \sum_a I_a^M \leq \beta x$. This two-sided cut defines a Boolean function $f : V_H \to \{0, 1\}$ with balance $p \leq 1/2$ satisfying $p \geq \min\{\beta/2, 1/3\} \geq \beta/3$ and $I^{M'_a} = I_i^f$. Using Lemma 2.4 with $\alpha = 2^{2x}$, we have

$$\frac{\beta x}{2^{2x}} + \sum_{a\in[d]} (I_a^{M'})^{4/3} \geq 2\frac{\beta}{3}\frac{2x}{2^{2x}}$$

We thus get $\sum_a (I_a^{M'})^{4/3} \geq \frac{\beta}{3}\frac{x}{2^{2x}}$. Now set $y = \max_{a\in[d]} I_a^{M'}$. Then, $\sum_a (I_a^{M'})^{4/3} \leq y^{1/3}\sum_a I_a^{M'} \leq \beta x y^{1/3}$. Therefore, we have $y^{1/3} \geq \frac{1}{3}2^{-2x}$, or, $y \geq 2^{-6x}/27$. □

The second lemma shows that if two functions $f, g : V_H \to \mathbb{R}$ are close in the sense that they agree on most of the inputs $v \in V_H$, then their influences are quite similar.

Lemma 2.5. *Let $H = (V_H, E_H)$ be a hypercube. If for two functions $f, g : V_H \to \mathbb{R}$ we have $f(v) = g(v)$ for all but a γ fraction of inputs $v \in V_H$, then for every dimension a we have $|I_a^f - I_a^g| \le 2\gamma$.*

Proof. Suppose that H is a d-dimensional hypercube, and consider a dimension-a edge $(u, v) \in E_H$. By our assumption, for all but at most $\gamma 2^d$ such edges, we must have $f(u) = g(u)$ and $f(v) = g(v)$, and in particular $f(u) - f(v) = g(u) - g(v)$. Recalling that there are exactly 2^{d-1} dimension-a edges, and that I_a^f is the fraction of those edges for which $f(u) - f(v) \neq 0$ (and similarly for g), we conclude that $f(u) - f(v) = g(u) - g(v)$ for at most 2γ fraction of the dimension-a edges, and thus $|I_a^f - I_a^g| \le 2\gamma$. $\square$

2.4 The NO instance

Lemma 2.6. *There exists $L = \Omega(\log 1/(\eta + \delta))$ such that if the* MULTICUT *instance G has a cutset of cost at most $2^d nL$ whose removal disconnects $\alpha \ge 7/8$ fraction of the demand pairs, then there exists a solution A for the unique 2-prover game G_Q whose value is larger than δ.*

Proof. Let $L = c \log 1/(\eta + \delta)$ where $c > 0$ is a constant to be determined later, and let $M \subseteq E$ be a cutset of cost $c(M) \le 2^d nL$ whose removal disconnects $\alpha \ge 7/8$ 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. Without loss of generality, we may assume that M is minimal with respect to containment, namely, for every subset $M' \subseteq M$, $M' \neq M$ (since removing M' from G disconnects fewer demand pairs than removing M would). Given such a minimal cutset M, for each cube C_i^p in G, consider the cutset M induces in this cube, and let $I_a^{p,i}$ be the influence of dimension $a \in [d]$ on this cutset. The randomized solution A (i.e., a strategy for the two provers) is defined as follows. For each vertex (question) $q_i^p \in Q$, we choose A_i^p to be the answer (dimension) $a \in [d]$ with probability $\frac{I_a^{p,i}}{\sum_a I_a^{p,i}}$.

We proceed to analyze the expected value of this randomized solution A. Recall that the value of a solution corresponds 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 $\{C_l^2\}_{l\in[n]}$. We will upper bound the probability of the following four "bad" events (for a choice of a question edge (q_i^1, q_i^2)):

$\mathcal{E}_1$ = fewer than half the demand pairs in C_i^1 are disconnected in $G \setminus M$.

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

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

$\mathcal{E}_4$ = M contains more than $2^d/2^{96L+7}$ cross edges between C_i^1 and C_j^2.

First, by our assumption above, removing M disconnects at least an $\alpha \ge 7/8$ fraction of the demand pairs inside the cubes $\{C_l^1\}_{l\in[n]}$, and thus by Markov's inequality, $\Pr[\mathcal{E}_1] \le 1/4$. Next, the cutset M contains at most $2^d nL$ hypercube edges, thus the expected number of edges in $C_i^1 \cup C_j^2$ that are contained in M is at most $2^d L$, and $\Pr[\mathcal{E}_2 \cup \mathcal{E}_3] \le 1/2$. Finally, if $c > 0$ is sufficiently small, $\Pr[\mathcal{E}_4] \le \eta L 2^{96L+7} \le \eta^{1/2} \le 1/8$, as otherwise the total cost along the corresponding question-edges (q_i^1, q_j^2) (i.e., those for which the cutset M contains more than $2^d/2^{96L+7}$ cross edges between C_i^1 and C_j^2) is more than $(\eta L 2^{96L+7}) \cdot (2^d/2^{96L+7}) \cdot (n/\eta) = n2^d L = c(M)$. Taking a union bound, the probability that any of the above bad events $\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3, \mathcal{E}_4$ occurs is less than $\frac{7}{8}$. In particular, if neither $\mathcal{E}_2$ nor $\mathcal{E}_3$ occurs, then $\sum_{a\in[d]} I_a^{1,i} \le 8L$ and $\sum_{a\in[d]} I_a^{2,j} \le 8L$.

In order to lower bound the expected value of the randomized solution A, we would like to show that if none of the above bad events happens, then there exists a dimension $a^* \in [d]$, such that in cube C_i^1 this dimension a^* has large influence, $I_{a^*}^{1,i}$, and in cube C_j^2 dimension $b_{ij}(a^*)$ has large influence, $I_{b_{ij}(a^*)}^{2,j}$. For the cube C_i^1, if the events $\mathcal{E}_1, \mathcal{E}_2$ do not occur, then we can use Lemma 2.3 (with $\beta = 1/2$, $x = 16L$) and conclude that there exists a dimension $a^* \in [d]$ such that

$$I_{a^*}^{1,i} \geq 2^{-96L}/27.$$

For the sake of analysis, label the connected components of $G \setminus M$ with distinct integer values. Define $f : C_i^1 \to \mathbb{Z}$ by letting $f(v)$ for $u \in C_i^1$ be the label of the connected component of u, and define $g : C_j^2 \to \mathbb{Z}$ similarly. For every $u \in C_i^1$, if $f(u) \neq g(b'_{ij}(u))$ then the cross edge $(u, b'_{ij}(u))$ must be contained in the cutset M, and because we assumed the event $\mathcal{E}_4$ happens, this occurs for at most $2^d/2^{96L+7}$ vertices $u \in C_i^1$. Applying Lemma 2.5 to the functions f and $g \circ b'_{ij}$, we conclude that $|I_a^f - I_a^{g \circ b'_{ij}}| \leq 2^{-96L-6}$ for all dimensions $a \in [d]$. Notice that for every edge (u, v) in the cube C_i^1, we have $f(u) = f(v)$ if and only if $(u, v) \notin M$, and for g in the cube C_j^2, and thus for all $a \in [d]$ we have $I_a^{1,i} = I_a^f$ and $I_a^{2,j} = I_a^g$. Since b_{ij} is just a permutation of the coordinates, for all we have $a \in [d]$, $I_a^{g \circ b'_{ij}} = I_{b_{ij}(a)}^g$. Altogether, we obtain

$$I_{b_{ij}(a^*)}^{2,j} \geq I_{a^*}^{1,i} - 2^{-96L-6} \geq 2^{-96L}/54,$$

and thus

$$\begin{aligned}
\Pr[A_j^2 = b_{ij}(A_i^1)] &\geq \Pr[A_i^1 = a^*, A_j^2 = b_{ij}(a^*)] \\
&\geq \frac{1}{8} \cdot \frac{I_{a^*}^{1,i}}{\sum_{a \in [d]} I_a^{1,i}} \cdot \frac{I_{a^*}^{2,j}}{\sum_{a \in [d]} I_a^{2,j}} \\
&\geq \Omega(L^{-2} 2^{-96L})
\end{aligned}$$

We conclude that the expected value of the randomized solution A is

$$\sum_{(i,j) \in E_Q} w(q_i^1, q_j^2) \Pr[A_j^2 = b_{ij}(A_i^1)] \geq \Omega(L^{-2} 2^{-96L}) > \delta,$$

where the last inequality holds if $c > 0$ is sufficiently small, and this completes the proof of Lemma 2.6. $\square$

2.5 Putting it all together

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

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 even NP-hard to obtain a $(7/8, L(N))$-bicriteria approximation. Note that the number of demand pairs is $k = n2^d = n^{\Theta(1)}$, and thus the hardness of approximation factor is similar when expressed in terms of k 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 previous section).

Lemma 3.1. *Let $0 < \alpha < 1$ be a constant. If there exists an algorithm for* SPARSEST-CUT *that produces in polynomial time a cut whose value is within factor $\rho \geq 1$ of the minimum, then there is a polynomial time algorithm that computes an $(\alpha, \frac{2\rho}{1-\alpha})$-bicriteria approximation for* MULTICUT.

Proof. Fix $0 < \alpha < 1$, and suppose $\mathcal{A}$ is an algorithm for SPARSEST-CUT that produces in polynomial time 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 $\{s_i, t_i\}_{i=1}^{p}$. We may assume without loss of generality that every s_i is connected (in G) to its corresponding t_i. Let $c_{\min}$ and $c_{\max}$ be the smallest and largest edge costs in G, and let $n = |V|$.

We now describe the bicriteria approximation algorithm for MULTICUT. For every value $C \in [c_{\min}, n^2 c_{\max}]$ which is a power of 2, execute a procedure that we will describe momentarily to compute a cutset $M_C \subseteq E$, and report, from all these cutsets M_C whose removal disconnects at least αk demand pairs, the one of least cost. For a given value $C > 0$, the procedure starts with $M_C = \emptyset$, and then iteratively M_C is "augmented" as follows: Take a connected component S of $G \setminus M_C$, apply algorithm $\mathcal{A}$ to $G[S]$ (the subgraph induced on S and all the demand pairs that lie inside S), and if the resulting cutset E_S has value (in $G[S]$) at most $\frac{\rho}{1-\alpha} \cdot \frac{C}{k}$, then add the edges E_S to M_C. Here, the value (ratio of cost to demands cut) of E_S is defined as $b_S = c(E_S)/|D_S|$, where D_S is the collection of demand pairs that lie in $G[S]$ and get disconnected (in $G[S]$) when E_S is removed. Proceed with the iterations until for every connected component S in $G \setminus M_C$ we have $b_S > \frac{\rho}{1-\alpha} \frac{C}{k}$, at which point the procedure returns the cutset M_C.

This algorithm clearly runs in polynomial time, so let us analyze its performance. We first claim that for every value C, the cutset M_C returned by the above procedure has sparsest-cut value (ratio of cost to demand disconnected, in G) is at most $\frac{\rho}{1-\alpha} \frac{C}{k}$. Indeed, suppose the procedure performs t augmentation iterations. Denote by S_i the connected component S that is cut at iteration $i \in [t]$, by E_{S_i} the corresponding cutset output by $\mathcal{A}$, and by D_{S_i} the corresponding set of demand pairs that get disconnected. Clearly, M_C is the disjoint union $E_1 \cup \cdots \cup E_t$, and it is easy to verify that the collection D_C of demand pairs cut by the cutset M_C is the disjoint union $D_{S_1} \cup \cdots \cup D_{S_t}$. Thus,

$$c(M_C) = \sum_{i=1}^{t} c(E_{S_i}) \leq \frac{\rho}{1-\alpha} \cdot \frac{C}{k} \sum_{i=1}^{t} |D_{S_i}| = \frac{\rho}{1-\alpha} \cdot \frac{C}{k} |D_C|,$$

which proves the claim.

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. The sparsest-cut value of M^* is $b^* = c(M^*)/k$. We will show that if $C \in [c(M^*), 2c(M^*)]$, then the above procedure produces a cutset M_C whose removal disconnects a collection D_C containing $|D_C| \geq \alpha k$ demand pairs; this will complete the proof of the lemma, because it immediately follows that

$$c(M_C) \leq \frac{\rho}{1-\alpha} \cdot \frac{C}{k} |D_C| \leq \frac{\rho}{1-\alpha} \cdot 2c(M^*),$$

and clearly $c(M^*) \in [c_{\min}, \binom{n}{2} \cdot c_{\max}]$. So suppose now $C \in [c(M^*), 2c(M^*)]$ and assume for contradiction that $|D_C| < \alpha k$. Denote by $V_1, \ldots, V_p \subseteq V$ the connected components of $G \setminus M_C$, and let D_j contain the demand pairs that lie inside V_j. It is easy to see that $\sum_{j=1}^{p} |D_j| = k - |D_C| > (1-\alpha)k$. Similarly, let M_j^* be the collection of edges in M^* that lie inside V_j. Then $c(M^*) \geq \sum_{j=1}^{p} c(M_j^*)$. Notice that, in every induced graph $G[V_j]$, the edges of M_j^* form a cutset (of $G[V_j]$) that cuts all the demand pairs in D_j. Using the stopping condition of the procedure, and since $\mathcal{A}$ provides an approximation within factor ρ, we have $c(M_j^*) \geq \frac{1}{1-\alpha}\frac{C}{k}|D_j|$ (the inequality is not strict because D_j might be empty). We thus derive the contradiction

$$c(M^*) \geq \sum_{j=1}^{p} c(M_j^*) \geq \frac{1}{1-\alpha} \cdot \frac{C}{k} \sum_{j=1}^{p} |D_j| > c(M^*).$$

This shows that when $C \in [c(M^*), 2c(M^*)]$, the procedure stops with a cutset M_C whose removal disconnects $|D_C| \geq \alpha k$ demand pairs, and concludes the proof of the lemma. $\square$

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 $\overline{u}$. There is a clause for every edge (u, v) in the graph G—$(x_u = x_v)$—with weight equal to w_e.

The following lemma is immediate from the construction and implies an analogue of Lemma 2.6 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—for any demand pair $(u, \overline{u})$, $S(x_u) \neq S(x_{\overline{u}})$. $\square$

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 for 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 $x_u = x_v$ 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 $2^d \frac{n}{\eta}\eta = 2^d n$.

The lemma follows from adding the two costs. $\square$

Lemmas 4.1 and 4.2 along with Lemma 2.6 imply the part of Theorem 1.2 regarding MIN-2CNF≡ DELETION.

5 Concluding remarks

The main bottleneck to improving the hardness factor lies in the in Lemma 2.3, which in turn crucially depends on Lemma 2.4, due to [KKL98]. These bounds are tight in general, as shown by the tribes function. But in our context, one may additionally assume that f is *odd*, that is, $f(x) \neq f(\overline{x})$ for all inputs x. Even with this additional assumption, this bound cannot be improved substantially, as demonstrated by the following variant of the tribes function: Partition the variables $x_1, \ldots, x_d$ into subsets of size $\log d - 2\log\log d$ each; the output is the value of the first unanimous subset, or x_1 if no unanimous tribe exists. This function is clearly odd, yet all variables have influence at most $O(\frac{\log^2 d}{d})$ and the total influence is $O(\log d)$. For Lemma 2.3, this function leads to a cutset M with $\beta = 1$, such that for $x = \sum_a I_a^M = O(\log d)$ we have $\max_a I_a^M \leq 2^{-\Omega(x)}$.

References

[ACMM04] A. Agarwal, M. Charikar, K. Makarychev, and Y. Makarychev. Private Communication, June 2004.

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

[ALN04] S. Arora, J. R. Lee, and A. Naor. Euclidean distortion and the sparsest cut. Manuscript, 2004.

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

[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*, 2004.

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

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

[CGR05] S. Chawla, A. Gupta, and H. Räcke. Improved approximations to sparsest cut. In *Proceedings of theAnnual ACM-SIAM Symposium on Discrete Algorithms*, 2005.

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

[DI03] E. Demaine and N. Immorlica. Correlation clustering with partial information. In *Proceedings of APPROX*, pages 1–13, 2003.

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

[EF03] D. Emanuel and A. Fiat. Correlation clustering—minimizing disagreements on arbitrary weighted graphs. In *Proceedings of ESA*, pages 208–220, 2003.

[FR04] U. Feige and D. Reichman. On systems of linear equations with two variables per equation. In *Proceedings of APPROX*, 2004.

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

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

[KARR90] P. Klein, A. Agrawal, R. Ravi, and S. Rao. Approximation through multicommodity flow. In *31st Annual Symposium on Foundations of Computer Science, Vol. I, II (St. Louis, MO, 1990)*, pages 726–737, 1990.

[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.

[KKL98] J. Kahn, G. Kalai, and N. Linial. The influence of variables on boolean functions. In *Proceedings of the 38th Annual IEEE Symposium on Foundations of Computer Science*, pages 68–80, 1998.

[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 Annual IEEE Symposium on Foundations of Computer Science*, 2004.

[KR03] S. Khot and O. Regev. Vertex cover might be hard to approximate to within $2 - \epsilon$. In *IEEE Conference on Computational Complexity*, pages 379–386, 2003.

[KV04] S. Khot and N. Vishnoi. On embeddability of negative type metrics into ℓ_1. Manuscript, 2004.

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

[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 *29th Annual Symposium on Foundations of Computer Science*, pages 422–431, October 1988.

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

[Ste00] D. Stefankovic. *Fourier Transforms in Computer Science*. PhD thesis, University of Chicago, 2000.

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 preserving its completeness and soundness. First we claim that we can assume that the ratio between the max weight $\max_e w_e$ and the min weight $\min_e w_e$ is bounded by n^3. This is because we can remove all edges with weight less than $\frac{1}{n^3} \max_e w_e$ from the graph, changing the soundness and completeness parameters by at most $\frac{1}{n}$. By a similar argument, we can assume that all weights in the graph are integral multiples of $t = \frac{1}{n^2} \min_e w_e$.

Now we convert 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 $w_e \frac{t}{W(1,i)} \frac{t}{W(2,j)}$.

Note that the total weight of all the edges remains the same as before. Each new node $q_i^1(x)$ 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. Therefore, the graph is regular. Furthermore, the number of vertices increases by a factor of at most n^6.

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 node $q_i^p(x)$ in Q' as the answer picked for q_i^p in Q. Likewise, consider a solution in Q'. Note that the answers for the questions $q_i^p(x)$ with different values of x must all be the same, because all these questions are connected to identical sets of vertices, with the same weights. Therefore, 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 given solution in Q'.

Thus for every solution in Q, there is a solution of the same weight in Q' and vice versa. This proves that the two games have exactly the same soundness and completeness parameters. $\square$