# Fourier analysis and inapproximability for MAX-CUT: a case study

Jake Wellens

May 6, 2016

**Abstract**

Many statements in the study of computational complexity can be cast as statements about Boolean functions $f : \{-1,1\}^n \to \{-1,1\}$. However, it was only very late in the last century that the *analytic* properties of such functions, often expressed via the Fourier transform on the Boolean hypercube, became the key ingredient in proofs of hardness of approximation. In this short survey, we give a brief overview of the history of this relationship between harmonic analysis and inapproximability for CSPs by zooming in on the particularly illustrative example of MAX-CUT. We summarize Hastad's seminal ideas from [4], proving unconditional NP-hardness of $(\frac{16}{17} + \epsilon)$-approximating MAX-CUT. Then we take a detailed look at how Khot, Kindler, Mossel and O'Donnell [7] pushed these Fourier-analytic methods further to prove UGC-hardness of approximating MAX-CUT to within any constant factor larger than $\alpha_{GW} \approx 0.878$, the factor achieved by the famous Goemans-Williamson approximation algorithm. In particular, we'll discuss the Majority is Stablest Theorem and the role it plays in their analysis, with the hope of making this connection between a purely analytic invariance principle and the surprising UGC-optimality of Goemans and Williamson's algorithm – as well as the need for analytic methods in proving computational hardness – appear a little less mysterious than it might at first sight.

## 1 Introduction: A brief history of MAX-CUT

MAX-CUT is a simple and classical problem in combinatorial optimization: given an undirected graph $G = (V, E)$ with edge weights $w_{ij} \geq 0$, the MAX-CUT problem[1] is to find a subset of vertices $C \subseteq V$ (called a cut) such that the combined weights of all the edges crossing the cut (i.e. with exactly one endpoint in $C$) is as large as possible. That is, MAX-CUT asks to maximize

$$\mathrm{val}(C) := \sum_{(i,j) \in E \cap C \times (V \setminus C)} w_{ij} \tag{1}$$

Given as one of Karp's original NP-complete problems in 1972, there has since been much interest in finding efficient algorithms for obtaining optimal and nearly-optimal solutions to

[1]Technically, finding an (approximately) optimal cut may be harder than finding the *value* of an (approximately) optimal cut. However, in all cases considered here, both problems are either easy or NP-hard, so we often ignore the distinction.

it[2]. Using a simple greedy algorithm, it's easy to find a cut in any graph with $\mathrm{val}(C) \geq \frac{1}{2}\sum_{(i,j)\in E} w_{ij}$. The first – and only – substantial improvement over this trivial approximation was made by Goemans and Williamson in 1994, whose randomized algorithm, described in section 3, obtains in expectation the suspiciously irrational approximation factor

$$\alpha_{GW} := \min_{\rho\in(-1,0)} \frac{2}{\pi}\frac{\arccos\rho}{1-\rho} = \min_{\theta\in(0,\pi)} \frac{2}{\pi}\frac{\theta}{1-\cos\theta} \approx 0.878 \tag{2}$$

The MAX-CUT problem is a special case of MAX-E2-Lin, which is the problem of finding an assignment to Boolean variables $x_i$ which satisfies as many linear constraints as possible, where each constraint has the form

$$x_i + x_j = b \mod 2; \quad \text{where } b \in \{0,1\} \tag{3}$$

(For MAX-CUT, of course, there is a constraint $x_i + x_j = 1 \mod 2$ for each edge $(i,j)$.) This is a typical example of a *constraint satisfiability problem* (or CSP), a class containing many other important hard problems like SAT and Vertex Cover. Finding optimal assignments for these and other CSPs has long been known to be NP-hard, but as the celebrated PCP theorem of Arora et al. [1] shows, there is a theoretical barrier to obtaining even *approximate* optimal solutions.

**PCP Theorem:** [Arora et al. 1998] *There is a universal constant $c < 1$ such that for any language $L \in NP$ and any string $x \in \{0,1\}^n$, we can construct in poly$(n)$-time a 3-CNF formula $\phi = \phi_{x,L}$ such that if $x \in L$, then $\phi$ is satisfiable; while if $x \notin L$, at most a fraction $c$ of $\phi$'s clauses are simultaneously satisfiable.*[3] *Furthermore, we can pick $\phi$ such that each clause contains exactly 3 distinct variables and each variable appears in $\phi$ exactly 5 times.*

After various versions of the PCP theorem established NP-hardness of approximating a variety of CSPs, it became a matter of great practical importance and intellectual interest to pinpoint these thresholds exactly. Proving lower bounds on such thresholds requires finding efficient algorithms (like Goemans-Williamson), while proving upper bounds (i.e. hardness of approximation theorems) usually requires constructing a particular type of PCP verifier whose tests and queries are suited to the CSP in question. Moreover, upper bounds for one CSP can often be turned into upper bounds for different CSPs via reduction gadgets, and this process was streamlined in [12]. We begin in Section 2 by taking a look at Hastad's Fourier-analytic approach [4] to proving hardness of approximation for CSPs, which in many cases, obtains either best-known or theoretically optimal hardness factors. In particular, he shows that $(\frac{16}{17} + \epsilon)$-approximating MAX-CUT is NP-hard, which, 15 years later, remains the best unconditional result in this direction. In Section 3, following the work of Khot, Kindler, Mossel and O'Donnell [7], we'll use similar techniques to improve this hardness factor as much as possible – right up to the $\alpha_{GW}$ threshold – assuming the (somewhat controversial) *Unique Games Conjecture.*

[2] In [2], it was shown that for MAX-CUT (as well as a number of other combinatorial constraint optimization problems), the optimal approximation factor for the weighted and unweighted (i.e. $w_{ij} = 1$) versions of the problem are the same up to some additive $o(1)$ term. Hence, for our purposes, we may use either one at any time.

[3] Henceforth we shall call this maximal fraction $\mathrm{val}(\phi)$, and use the same notation for the analogous fractions in other CSPs.

## 2 Hastad's NP-Hardness for $(\frac{16}{17}+\epsilon)$-approximating MAX-CUT

Unlike the approach of Khot et al. [7], Hastad first shows hardness of MAX-E3-Lin and then uses a gadget of Trevisan et al. [12] to reduce from MAX-E3-Lin to MAX-CUT. Specifically, Hastad constructs a PCP for an NP-hard language with completeness $1-\epsilon$ and soundness $\frac{1}{2}+\epsilon$ for any $\epsilon > 0$, making only 3 queries to the proof and a test of the form $x_i + x_j + x_k =^? b \mod 2$.

### 2.1 Two-Prover One-Round Games and the Label Cover problem

Hastad's proof of soundness is based on a simple protocol known as a *two-prover one-round game*. We will describe the relevant protocol below in a style similar to Hastad's original presentation, but we will quickly shift our model to that of *label cover* problems. These two frameworks are equivalent, but the latter is now more common in the literature and, for our purposes, more amenable to analogy with the PCP of Khot et al. that we will encounter in section 2.

**Basic two-prover protocol:**

*Input:* A 3-CNF formula $\phi = C_1 \wedge \cdots \wedge C_m$, where $C_j = (x_{j_1} \vee x_{j_2} \vee x_{j_3})$.

*Protocol:* 1) Pick $j \in [m]$ and $k \in \{j_1, j_2, j_3\}$ uniformly at random and send $j$ to $P_1$ and $k$ to $P_2$.

2) Receive values for $x_{j_1}, x_{j_2}, x_{j_3}$ from $P_1$ and $x_k$ from $P_2$. Accept iff the two values for $x_k$ agree and $C_j$ is satisfied.

**Claim:** *If val$(\phi) = c$, then the above protocol accepts $\phi$ with probability at most $(2+c)/3$.*

**Proof:** The answers that $P_2$ gives define an assignment $y$ to all the variables. If a clause $C$ is unsatisfied by $y$, then in order for $P_1$'s responses to satisfy $C$, it must disagree with $P_2$ on at least 1 of the 3 variables in that clause. Thus the test rejects with probability at least $(1-c)/3$. □

Since the value of $c < 1$ from the PCP theorem (which will translate to soundness $(2+c)/3$ in the above protocol) may be very close to 1, we'll want to amplify this gap[4] via *parallel repetition*:

**$r$-parallel two-prover protocol:**

*Input:* A 3-CNF formula $\phi = C_1 \wedge \cdots \wedge C_m$, where $C_j = (x_{j_1} \vee x_{j_2} \vee x_{j_3})$.

*Protocol:* 1) For $l = 1, \ldots, r$, pick $j^l \in [m]$ and $k^l \in \{j_1^l, j_2^l, j_3^l\}$ uniformly and independently and send all $j^l$ to $P_1$ and all $k^l$ to $P_2$.

2) Receive values for $x_{j_1^l}, x_{j_2^l}, x_{j_3^l}$ from $P_1$ and $x_k^l$ from $P_2$. Accept iff the two values for $x_{k^l}$ agree and $C_j$ is satisfied for every $l = 1, \ldots, r$.

Thus, it follows from

[4] Of course, if $\phi \in$ SAT, then honest provers can get the test to accept with probability 1.

**Parallel Repetition Theorem:** [Raz 1998] *For any $d = O(1)$ and $s < 1$, there is a constant $\tau = \tau(d, s) < 1$ such that given a two-prover one-round proof system with soundness $s$ and answer size*[5] *at most $d$, the soundness of the $r$-parallel version of the protocol is at most $\tau^r$.*

that for $\epsilon > 0$ and $r = r(\epsilon)$ sufficiently large[6], the above protocol accepts a formula $\phi$ coming from the PCP theorem with probability 1 if $\text{val}(\phi) = 1$ and probability at most $\epsilon$ if $\text{val}(\phi) \leq c$.

We now move to the equivalent setting of Label-cover:

**Definition:** The *Label-Cover problem with alphabet* $\Sigma$ is defined by the tuple $(G, V_1, V_2, E, \{\pi_e\}, \Sigma)$, where

- $G = (V_1, V_2, E)$ is bipartite and each vertex in $V_1$ has the same degree
- The constraints $\pi_e$ for each edge $e = (u, v) \in E$ are functions $\pi_e : \Sigma \to \Sigma \cup \{\#\}$ which are satisfied iff the labels $a_u, a_v \in \Sigma$ given to $u$ and $v$ satisfy $\pi_e(a_u) = a_v$.

Thinking of $V_1$ as the set of questions for the first prover $P_1$, and $V_2$ as the set of questions for $P_2$, and the functions $\pi_e$ as encoding the predicate the verifier uses to accept/reject based on the provers' responses, one can easily convince oneself that these two models capture the same protocol: an assignment of "labels" to vertices in the latter is simply a choice of "answers" to questions in the former, and the fraction of edge constraints which are satisfied by an assignment equals the probability the verifier accepts when the provers use the corresponding strategy.[7] For concreteness, Hastad's basic protocol from the previous page translates to the Label-Cover instance with $V_1$ equal to the set of clauses $C_i$ in $\phi$, and $V_2$ is equal to the set of variables $x_j$ in $\phi$. The alphabet $\Sigma$ will be $\{0, 1\}^3 \cup \{0, 1\}$, and the edge constraint $\pi_{(u,v)}$ will send the subset of $\{0, 1\}^3$ that satisfies a clause $C_u$ to the assignment of the $x_v$ variable, and send everything else to $\#$.

Let $\text{GAP-LC}_{1,\epsilon}(\Sigma)$ be the problem of distinguishing Label-Cover instances which are 1-satisfiable from those which are at most $\epsilon$-satisfiable. By translating our amplified (parallelized) protocol into a Label Cover problem and using the PCP theorem, we have

**Theorem:** *For any $\epsilon > 0$, there is a finite alphabet $\Sigma$ such that* $\text{GAP-LC}_{1,\epsilon}(\Sigma)$ *is NP-hard.*

Hence, to show $s/c$-hardness for MAX-E3-Lin, it suffices to construct a PCP for $\text{GAP-LC}_{1,\epsilon}(\Sigma)$ (with soundness $s$ and completeness $c$) which only queries 3 bits and checks the parity of their sum.

## 2.2 Hastad's PCP for GAP-LC$_{1,\epsilon}(\Sigma)$

Let $\Sigma = \{1, \ldots, m\}$. As a first try, our verifier could simply pick a random edge $(u, v) \in E$, ask for the labels $a(u)$ and $a(v)$ and just check if $\pi_{(u,v)}(a_u) = a_v$. However, these labels belong

[5] Rambunctious provers can always troll the verifier with arbitrarily long junk answers, but here we only consider the size of answers that could possibly cause the verifier to accept.

[6] One can take $r = O(\log 1/\epsilon)$.

[7] For non-uniform distributions, one needs to use a *weighted* version of Label-Cover, but we ignore this technicality.

to $\{1, \ldots, m\}$, so it's not clear how to turn this predicate into a linear equation over $\mathbb{F}_2$. As is common in the construction and analysis of certain PCPs, it is convenient to work with a particular boolean encoding of assignments called *the long code*. Given $a \in \{1, \ldots, m\}$, the **long code**[8] of $a$ is just the string of length $2^m$ with entires indexed by $x \in \{-1, 1\}^m$ such that the $x$ entry of the string is $x_a$. We also use the notation[9] $x \circ \pi$ to mean the string $(x_{\pi(1)}, \ldots, x_{\pi(m)})$, for $x = (x_1, \ldots, x_m) \in \{-1, 1\}^m$ and $\pi : [m] \to [m]$.

**Hastad's PCP:** The verifier expects the proof to contain the list of long codes of the assignments to the vertices of $G$. She picks an edge $e = (u, v) \in E$ uniformly at random, and two independent, uniform random strings $x, y \in \{-1, 1\}^m$. Also, select a random $\mu \in \{-1, 1\}^m$ with an $\epsilon$-bias, that is,

$$\mu_i = \begin{cases} 1 \text{ with probability } 1 - \epsilon \\ -1 \text{ with probability } \epsilon \end{cases}$$

Letting $f$ and $g$ be the supposed long codes of $a(v)$ and $a(u)$, respectively, the verifier checks the condition

$$f(x)g(y)g((x \circ \pi)y\mu) = 1 \tag{4}$$

*Remark*: We need to make our verifier just a little bit more intelligent – if instead of a legitimate long code the prover just sends all 1's, the test (4) will always pass and the verifier will accept. We do want to make it unlikely for the verifier to accept something that isn't a long code, but we can't afford to add any additional checks and still extract optimal hardness for E3-Lin. As we will see, however, as long as the proof is *balanced* (i.e. contains the same number of 1's and -1's), this test is already pretty good at weeding out fake long codes. The easy fix, which Hastad calls *folding over True*, is to simply *pretend* the code is balanced in exactly the way a long code would be – for each pair $\{x, -x\}$, fix some choice for $x$ and whenever the $(-x)$-entry of a code is needed, just query the $x$-entry and negate the answer. This is the full description of Hastad's PCP; it remains to analyze the soundness and completeness parameters.

**Completeness:** If not for the biased noise $\mu$, the condition (4) would always hold if $f$ and $g$ were actually the long codes of $a(v)$ and $a(u)$, and the edge constraints $\pi_e(a(u)) = a(v)$ were satisfied. Indeed, in this case, for any $x, y$, $f(x) = x_{a(v)}$ and $g(y) = y_{a(u)}$, while

$$g((x \circ \pi)y\mu) = x_{\pi(a(u))}y_{a(u)}\mu_{a(u)} \tag{5}$$

and hence (4) is only violated when $\mu_{a(u)} = -1$, which happens with probability $\epsilon$. Hence, the completeness is $1 - \epsilon$.

**Soundness:** As always, soundness is more involved and this is where the Fourier analysis comes in.

[8] Thinking of $a$ as an element of $\{-1, 1\}^{\log m}$, and strings $x \in \{-1, 1\}^m$ as (truth tables of) functions $f : \{-1, 1\}^{\log m} \to \{-1, 1\}$, then the long code of $a$ simply lists $f(a)$ for all such functions $f$.

[9] Our notation and style throughout this section is based more directly on the lecture notes (http://courses.cs.washington.edu/courses/cse533/05au/) by Ryan O'Donnell than on Hastad's original exposition, although what's actually happening is entirely the same as in the original paper.

**Definitions:** *(Fourier analysis)* Given a subset $S \subseteq [m]$, define the linear parity function $\chi_S : \{-1,1\}^m \to \{-1,1\}$ to be

$$\chi_S(x) = \prod_{i \in S} x_i$$

We define the inner product $\langle f, g\rangle$ of two functions on $\{-1,1\}^m$ to be $\frac{1}{2^m}\sum_{x\in\{-1,1\}^m} f(x)g(x)$, and it's not hard to see that the parity functions $\chi_S$ satisfy

$$\langle \chi_S, \chi_T\rangle = \frac{1}{2^m}\sum_{x\in\{-1,1\}^m} \chi_{S\Delta T}(x) = \begin{cases} 0 \text{ if } S \neq T \\ 1 \text{ else} \end{cases} \tag{6}$$

and hence they form a complete orthonormal basis for the space of functions on $\{-1,1\}^m$. Accordingly, we define the *Fourier coefficients* of a function $f$ to be $\hat{f}(S) = \langle f, \chi_S\rangle$. In particular, if the truth table of $f$ is balanced, then $\hat{f}(\emptyset) = 0$. Moreover, since $\langle f, f\rangle = 1$ for Boolean-valued functions, we have Parseval's formula $1 = \sum_S \hat{f}(S)^2$. We'll also want to use the notation

$$\tau(S) := \{j \in [m] \text{ such that } |\pi^{-1}\{j\} \cap S| \text{ is odd}\} \tag{7}$$

since then $\chi_S(x \circ \pi) = \chi_{\tau(S)}(x)$.

Now we can use the basic tools of Fourier analysis to derive a useful expression for the probability of Hastad's verifier accepting. Conditioning on a particular choice of edge $(u, v)$ we have

$$\Pr[acc|(u,v)] = \frac{1}{2} + \frac{1}{2}\mathbb{E}_{u,v,x,y,\mu}[f(x)g(y)g((x\circ\pi)y\mu)] \tag{8}$$

Writing $f$ and $g$ (again, viewed as truth tables of functions) in terms of their Fourier expansions, we can expand

$$\mathbb{E}_{u,v,x,y,\mu}[f(x)g(y)g((x\circ\pi)y\mu)] = \sum_{S,T,U} \hat{f}(S)\hat{g}(T)\hat{g}(U)\mathbb{E}_{x,y,\mu}[\chi_S(x)\chi_T(y)\chi_U((x\circ\pi)y\mu)]$$

Here we observe that $\chi_U((x\circ\pi)y\mu) = \chi_{\tau(U)}(x)\chi_U(y)\chi_U(\mu)$ and use independence to rewrite the sum as

$$\sum_{S,T,U} \hat{f}(S)\hat{g}(T)\hat{g}(U)\mathbb{E}_x[\chi_S(x)\chi_{\tau(U)}(x)]\mathbb{E}_y[\chi_T(y)\chi_U(y)]\mathbb{E}_\mu[\chi_U(\mu)] \tag{9}$$

By independence of $\mu$'s entries, we see $\mathbb{E}_\mu[\chi_U(\mu)] = \prod_{i\in U}\mathbb{E}[\mu_i] = (1-2\epsilon)^{|U|}$. By orthogonality, $\mathbb{E}_x[\chi_S(x)\chi_{\tau(U)}(x)] = \langle \chi_S, \chi_{\tau(U)}\rangle = \delta_{S,\tau(U)}$ and similarly $\mathbb{E}_y[\chi_T(y)\chi_U(y)] = \delta_{T,U}$. Thus, the only terms which don't contribute zero to the sum occur when $T = U$ and $S = \tau(T)$, and so the sum becomes

$$\sum_T \hat{f}(\tau(T))\hat{g}(T)^2(1-2\epsilon)^{|T|} \tag{10}$$

Thus, if $\Pr[acc] \geq \frac{1}{2} + \delta$, then

$$\mathbb{E}_{(u,v)}[\sum_T \hat{f}(\tau(T))\hat{g}(T)^2(1-2\epsilon)^{|T|}] \geq 2\delta \tag{11}$$

and thus (since the sum inside the expectation is always at most 1), for at least a $\delta$ fraction of edges $(u, v)$, we have

$$\sum_T \hat{f}(\tau(T))\hat{g}(T)^2(1-2\epsilon)^{|T|} \geq \delta \tag{12}$$

Let us call such an edge a *good edge*. We need to somehow turn this observation into an assignment of labels to the vertices in our Label Cover instance which satisfies a decent portion of the edge constraints. To show the existence of such a labeling, it suffices to show that a *random* labelling has this property when we chose our labels according to some (cleverly constructed) distribution. Hastad's main idea is this: **we should choose labels which contribute the most to the mass of the Fourier coefficients $\hat{f}(S), \hat{g}(S)$ from the proof string**. More precisely, for each vertex $u \in V_1$, pick a set $S \subset [m]$ with probability $\hat{g}(S)^2$, and then choose $u$'s label $a(u)$ uniformly at random from $S$; we do the same for $v \in V_2$ using the Fourier coefficients of $f$ instead.[10] Thus, the probability that $u$ gets the label $a \in [m]$ is

$$\sum_{S \ni a} \hat{g}(S)^2 \frac{1}{|S|} \tag{13}$$

From this expression (which is closely related to an analytic property of Boolean functions called *influence* that will play an important role in Section 3), we can see that this distribution does a good job of favoring the "important" labels; indeed, if $g$ were (very close to) the long code of the label $a$, then we would pick $a(u) = a$ with probability (very close to) 1. Moreover, labeling vertices at random in this way, we have that for any edge $(u, v)$, the probability of the event $\pi(a(u)) = a(v)$ is at least

$$\sum_S \hat{f}(\tau(S))^2 \hat{g}(S)^2 \frac{1}{|S|} \tag{14}$$

since if we label $u$ from $S$, then by selecting $\tau(S)$ as our subset to label $v$, we automatically know that we'll succeed with probability at least $1/|S|$, as every element of $\tau(S)$ has an odd (and hence non-zero!) number of $\pi$-preimages in $S$. This expression already bears a striking resemblance to the expression in (12), which we know is at least $\delta$ for the good edges. Indeed, using only a bit of standard analysis[11], Hastad is able to show[12]

$$\sum_S \hat{f}(\tau(S))^2 \hat{g}(S)^2 \frac{1}{|S|} \geq 4\epsilon \left( \sum_T \hat{f}(\tau(T))\hat{g}(T)^2(1-2\epsilon)^{|T|} \right)^2 \tag{15}$$

which is, for good edges, at least $4\epsilon\delta^2$, and since at least $\delta$ of the edges are good, our random assignment will satisfy at least $4\epsilon\delta^3$ edge constraints. Thus, if our Label-Cover instance were only $\epsilon' < 4\epsilon\delta^3$ satisfiable, Hastad's verifier must accept with probability less than $1/2 + \delta$.

[10] We never choose $S = \emptyset$ because of folding ($f$ balanced $\implies \hat{f}(\emptyset) = 0$), and our probabilities add up to 1 by Parseval's identity.

[11] We omit the details, but this basically follows from the Cauchy-Schwartz inequality and the simple exponential inequality $1/|S| \geq 4\epsilon(1-2\epsilon)^{2|S|}$.

[12] In this step, we can see one reason why it was essential to introduce the noise $\mu$ – otherwise, Fourier coefficients of large sets $S$ would receive no "penalty" $(1-2\epsilon)^{|S|}$, which was needed to lower bound the probabilistic penalty $\frac{1}{|S|}$.

Since GAP-LC$_{1,\epsilon}(\Sigma)$ is NP-hard for any $\epsilon' > 0$, we have established[13] the main theorem of this section and of Hastad's paper:

**Theorem:** [Hastad, 2001] *For any $\epsilon > 0$, GAP-E3-Lin$_{1-\epsilon,\frac{1}{2}+\epsilon}$ is NP-hard. In particular, there is no efficient $(1/2+\epsilon)$-approximation algorithm for MAX-E3-Lin unless P=NP.*

Finally, we remark that both of the parameters in this theorem are optimal for GAP-E3-Lin$_{1-\epsilon,\frac{1}{2}+\epsilon}$, unless P=NP. Indeed, negating an assignment to an E3-Lin instance flips the value of each constraint, and so the better of any assignment and its negation is always a 1/2-approximation; on the completeness side, one can use Gaussian elimination to determine if a linear system over any field $\mathbb{F}$ has a solution in polynomial time.

## 2.3 From MAX-E3-Lin to MAX-CUT with an optimal gadget

To translate MAX-E3-Lin hardness to MAX-CUT, we use a reduction technique based on the familiar idea of *gadgets*. Gadgets have long been ubiquitous tools in hardness of approximation (and computer science in general), and while studying a particular gadget is usually not so enlightening, we use the section as an opportunity to touch upon an important paper of Trevisan, Sorkin, Sudan and Williamson [12]. In this paper, the authors formally define the notion of an $\alpha$-gadget, and give an efficient, linear programming based method for not only finding gadgets, but also, remarkably, proving bounds for the *best possible* gadget reducing one CSP to another.

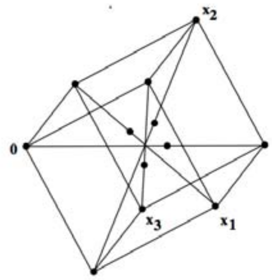
**Definition:** [12] Given a constraint function $f : \{0,1\}^k \to \{0,1\}$ and a constraint family $\mathcal{F} = \{C_1, \ldots, C_m\}$ over primary variables $X_1, \ldots, X_k$ and auxilliary variables $Y_1, \ldots, Y_n$, we define a **(strict)-$\alpha$-gadget** $\Gamma = (Y, C, w)$ to be a collection of non-negative weights $w_i$ such that, for boolean assignments $x = (x_i)_{i=1}^k$ to the $X_i$ and $y = (y_i)_{i=1}^n$ to the $Y_i$, we have

$$f(x) = 1 \quad \Longrightarrow \quad \max_{y \in \{0,1\}^n} \sum_{j=1}^m w_j C_j(x,y) = \alpha \tag{16}$$

$$f(x) = 0 \quad \Longrightarrow \quad \max_{y \in \{0,1\}^n} \sum_{j=1}^m w_j C_j(x,y) = \alpha - 1 \tag{17}$$

It's not hard to see that when it comes to transferring hardness of approximation from one problem to another, a *smaller* $\alpha$ means a better hardness result – a gadget achieving the minimal $\alpha$ among all gadgets from $f \to \mathcal{F}$ is therefore called *optimal*. Trevisan et al. construct a strict and optimal 8-gadget from the constraint function $f = f(x_1, x_2, x_3) = 1 + x_1 + x_2 + x_3 \mod 2$ to the constraint family $\mathrm{CUT}_0 := \{\mathrm{CUT}, id\}$, where $\mathrm{CUT}(x_1, x_2) = x_1 + x_2 \mod 2$, as well as a strict and optimal 9-gadget from $1 - f$ to $\mathrm{CUT}_0$. The inclusion of the identity constraint in $\mathrm{CUT}_0$ is not of any consequence for MAX-CUT – by adding a special vertex called 0, we can replace the condition $id(X_i) = 1$ with $\mathrm{CUT}_0(X_i, 0) = 1$ and maintain the same number of satisfied constraints[14].

[13] Technically, to show NP-hardness we need to describe a *reduction* from GAP-LC$_{1,\epsilon}(\Sigma)$. This is routine and essentially amounts to showing that the PCP we constructed can only ask about polynomially many E3-Lin




The graph shown above, with all implied edge weights 0.5, is an optimal 8-gadget reducing $f(x) = x_1 + x_2 + x_3 + 1$ to $\mathrm{CUT}_0$: that is, if $f(x) = 1$, then there is a way to extend this assignment to a cut on the gadget (which assigns 0 to 0) with weight 8, while if $f(x) = 0$, then the best possible extension yields a cut of weight 7. After constructing a similar 9-gadget[15] for $1 - f(x)$, we have a reduction from MAX-E3-Lin to MAX-CUT, given by building a graph $G$ out of these gadgets corresponding to each equation in an E3-Lin system $\Pi$. A simple counting argument shows that, for any $\epsilon > 0$, a cut on $G$ with weight at least $\frac{16}{17} + \epsilon$ times the weight of an optimal cut in $G$ corresponds to an assignment satisfying at least $\frac{1}{2} + \delta$ times the optimal number of equations satisfiable in $\Pi$ (for some $\delta = \delta(\epsilon) > 0$). Thus, we obtain Hastad's hardness result:

**Theorem:**[Hastad, 2001] *MAX-CUT is NP-hard to approximate within a factor of* $\frac{16}{17} + \epsilon$, *for any* $\epsilon > 0$.

In [4], Hastad uses similar arguments for many other CSPs besides MAX-CUT, of course, and many of these results are provably optimal. Almost always the algorithms achieving these thresholds are quite simple and natural, which seems to agree with the intuition that if an algorithm needs to do a bunch of circuitous and complicated things, the space of possible improvements becomes huge, and so such an algorithm is unlikely to be optimal.[16] For MAX-CUT however, the naive greedy algorithm and its modifications can only achieve $\frac{1}{2} + o(1)$ approximations, which is quite far from Hastad's upper bound. Up next, we'll describe an approximation for MAX-CUT by Goemans and Williamson, which, in light of the work of Khot et al. [7] and Raghavendra [11], may need to be thought of as the "natural" efficient approximation algorithm for MAX-CUT.

---

equations.

[14] Essentially, this just allows us to have one special reference vertex, which can always assumed to get the value 0 (just negate the whole assignment otherwise, which doesn't change the cut).

[15] The 9-gadget can be obtained from the 8-gadget by replacing the 0 vertex by a vertex $Z$, and adding an edge of weight 1 from 0 to $Z$.

[16] For example, the simple greedy algorithm for MAX-E3-SAT which just maximizes the expected number of clauses satisfied by the remaining variables achieves the optimal (up to $o(1)$) approximation factor of 7/8.

## 3 UGC-Hardness for $(\alpha_{GW} + \epsilon)$-approximating MAX-CUT

### 3.1 The Goemans-Williamson Algorithm for MAX-CUT

Recall that the weight of a cut (thought of as a $\{-1, 1\}$-assignment of the $x_i$ variables) is

$$\sum_{ij} \frac{w_{ij}(1 - x_i x_j)}{2}$$

and thus MAX-CUT becomes the integer quadratic program

$$\max_{x_i = \pm 1} \sum_{ij} \frac{w_{ij}(1 - x_i x_j)}{2} \tag{18}$$

While integer programming is likely intractable, the above problem admits a natural semi-definite relaxation, which forms the basis of the Goemans-Williamson algorithm. Without further ado, here is a description and analysis of their algorithm:

**The Goemans-Williamson Algorithm:** Let $S^{n-1} \subset \mathbb{R}^n$ denote the unit sphere. Using standard algorithms for convex optimization (e.g. the ellipsoid method), efficiently obtain (to arbitrary precision) an optimal solution $\{v_1, \ldots, v_n\}$ to the semi-definite program

$$\max_{v_i \in S^{n-1}} \sum_{ij} \frac{w_{ij}(1 - \langle v_i, v_j \rangle)}{2} \tag{19}$$

which is evidently a relaxation of (18). Then pick a uniformly random vector $a \in S^{n-1}$ and set $x_i = \operatorname{sgn}(\langle a, v_i \rangle)$. The probability that $x_i \neq x_j$ is, by the geometry of the sphere, proportional to the angle between $v_i$ and $v_j$, and so their expected contribution to the cut is $\frac{w_{ij}}{\pi} \arccos \langle v_i, v_j \rangle$. Thus the $\alpha_{GW}$ factor arises from computing the worst-case ratio of this contribution to the corresponding contribution in (19).

Despite the reasonable simplicity of this algorithm, there doesn't seem to be anything obviously *optimal* about it. While the integrality gap for the SDP (19) has since been proven to be exactly $\alpha_{GW}$, this doesn't rule out the possibility of improvement by adding constraints[17] to the SDP and modifying the rounding scheme, as has been exploited to achieve better approximation ratios for special classes of graphs. This is why the result of Khot et. al. [7] comes as something of a surprise. As we'll see, however, the analysis behind the Majority is Stablest theorem [9] has something to say about the fundamental geometry behind Goemans-Williamson's optimality. Indeed, an even stronger invariance principle was used by Raghavendra in 2008 [11] to prove a remarkably general result in this direction: *for essentially every CSP, the integrality gap of a natural SDP relaxation of the CSP is equal to its inapproximability threshold, assuming the Unique Games Conjecture.*

[17] One common modification is to add the triangle inequality constraints: $\|v_i - v_j\|^2 + \|v_j - v_k\|^2 \geq \|v_i - v_k\|^2$. Khot and Vishnoi [8] were able to show that this less-relaxed relaxation of MAX-CUT still has an integrality gap of $\alpha_{GW}$, however.

### 3.2 Unique Label Cover and the UGC

Looking back at the construction of Hastad's E3-Lin-based PCP for GAP-LC$_{1,\epsilon}(\Sigma)$, one might wonder if we could modify the test in such a way that we could obtain MAX-CUT hardness *directly* from Label Cover hardness, as Hastad does for a variety of other $k$-CSPs, for $k \geq 3$. As it turns out, when $k = 2$, the bottleneck in such attempts comes from the fact that the projections $\pi_e : \Sigma \to \Sigma$ in a Label Cover instance can fail to be *injective*, and thus Khot suggested using *Unique Label Cover* as a starting point in proving hardness for 2-CSPs. Unique Label Cover, as Khot defined it, is the restricted form of Label Cover in which all edge constraints $\pi_e$ are actually *permutations*. The attentive reader may have noticed that Gap-ULC$_{1,1-\epsilon}(\Sigma)$ is actually in P – indeed, the label of one vertex uniquely determines the labels of all other vertices in its connected component. Thus, Khot put forth the following conjecture, which is, in a sense, the best hardness statement we could dream of for Unique Label Cover:

**The Unique Games Conjecture:** *For any $\epsilon, \delta > 0$, there is a constant $M = M(\epsilon, \delta)$ such that Gap-ULC$_{1-\delta,\epsilon}(\Sigma)$ is NP-hard for $|\Sigma| \geq M$.*

Since its introduction in 2002, this conjecture has become a central figure in hardness of approximation, yet there is still a lack of convincing evidence for either its truth or its falsehood. However, numerous algorithms have been discovered which show certain relationships that the parameters in the UGC must obey, if it is to hold. A survey by Khot [6] contains a nice summary of recent results in this direction. Of course, at this point *any* hardness result (not just NP-hardness) for Unique Label Cover will have important algorithmic implications, and thus the truth of UGC is perhaps less important than how *far* from true it is.

### 3.3 Influence, Noise Stability, and Majority-is-Stablest

To prove UGC-Hardness of MAX-CUT, Khot et al. construct a PCP for Unique Label Cover which, in a vague sense, is the 2-query analogue of Hastad's 3-query PCP for Label Cover. The PCPs and their soundness/completeness analysis resemble one another closely, but establishing soundness in the 2-query case requires a much deeper result from the analysis of Boolean functions called the Majority Is Stablest theorem, which we state below after introducing a bit of relevant terminology.

**Definitions:** *(More Fourier analysis)* The *influence* of coordinate $i$ on a function $f : \{-1,1\}^n \to \{-1,1\}$ is defined as

$$\mathrm{Inf}_i(f) := \Pr[f(x_1, \ldots, x_i, \ldots, x_n) \neq f(x_1, \ldots, -x_i, \ldots, x_n)] = \sum_{S \ni i} \hat{f}(S)^2 \tag{20}$$

Inspired by the second equality, we also define the *$k$-degree influence* of coordinate $i$ to be

$$\mathrm{Inf}_i^k(f) = \sum_{S \ni i, |S| \leq k} \hat{f}(S)^2$$

To build some intuition, the reader should check that

$$\mathrm{Inf}_i^k(\chi_S) = \begin{cases} 1 \text{ if } i \in S \text{ and } |S| \leq k \\ 0 \text{ else} \end{cases} \qquad \mathrm{Inf}_i(\mathrm{OR}_n) = \mathrm{Inf}_i(\mathrm{AND}_n) = 2^{1-n}$$

$$\mathrm{Inf}_i(\mathrm{MAJ}_n) = \binom{n-1}{\frac{n-1}{2}} 2^{1-n} = \frac{\sqrt{2/\pi}}{\sqrt{n}} + O(n^{-3/2}) \ \ \text{(for odd } n\text{)}$$

A related notion is *noise stability* of $f$, which, rather than flipping one bit and measuring how likely it is that $f$ changes, flips a certain "fraction" of bits at random. More precisely, let $\rho$ be a fixed parameter in $[-1, 1]$, let $x$ be a uniformly random string in $\{-1, 1\}^n$ and let $y$ be $\rho$-correlated copy of $x$, that is, set $y_i = x_i$ with probability $\frac{1}{2} + \frac{\rho}{2}$, and $y_i = -x_i$ otherwise. Then the noise stability of $f$ at $\rho$ is defined to be[18]

$$S_\rho(f) = \mathbb{E}_{x,y}[f(x)f(y)] = 2\Pr[f(x) = f(y)] - 1 = \sum_S \rho^{|S|} \hat{f}(S)^2$$

Observe that dictator functions $\chi_i$ have noise stability $\rho$, and more generally, all parity functions have $S_\rho(\chi_S) = \rho^{|S|}$. Using the Central Limit Theorem, it isn't too hard to show that the asymptotic noise stability of the majority function $\mathrm{MAJ}_n : \{-1, 1\}^n \to \{-1, 1\}$ obeys the formula

$$\lim_{n\to\infty} S_\rho(\mathrm{MAJ}_n) = 1 - \frac{2 \arccos \rho}{\pi} \tag{21}$$

We can recast the noise stability of a function $f$ in terms of the probability that the following "$\neq$" test accepts: choose $x \in \{-1, 1\}^n$ uniformly at random, and choose $\mu \in \{-1, 1\}^n$ such that $\mu_i = -1$ with probability $\frac{1-\rho}{2}$ and 1 otherwise; then test $f(x) \neq f(x\mu)$. The probability such a test accepts is of course $\frac{1}{2} - \frac{1}{2}S_\rho(f)$. Hence, the probability this test accepts a dictator is $\frac{1}{2} - \frac{1}{2}\rho$, while the probability it accepts $\mathrm{MAJ}_n$ is very close to $\frac{\arccos \rho}{\pi}$ for large $n$. What can we say about this probability for general $f$? It turns out that when $f$ has small influences, it won't pass this test with probability significantly greater than $\mathrm{MAJ}_n$ – this is essentially the content of the Majority is Stablest theorem.

Before stating the precise theorem we need, we'll motivate it with a heuristic analysis in the case when $f : \{-1, 1\}^n \to \{-1, 1\}$ is a half-space function, i.e. $f(x) = \mathrm{sgn}(a \cdot x)$, where we can rescale to assume $\sum_i a_i^2 = 1$. It's not so difficult to check that $f$ having "small influences" means each $a_i$ is fairly small. Then, as $a \cdot x$ is a linear combination of i.i.d. random variables with small coefficients, a strong form of the Central Limit Theorem implies that the distribution of $a \cdot x$ will be roughly Gaussian. Setting $b = a\mu$, then $f(x\mu) = b \cdot x$, which will therefore also be roughly Gaussian, conditioned on a fixed choice of $\mu$. Moreover, a 2-dimensional version of the Central Limit Theorem tells us that the joint distribution of $a \cdot x$ and $b \cdot x$ will be close to the joint distribution of $a \cdot g$ and $b \cdot g$, where $g$ is a vector with independent mean-zero Gaussian entries. By symmetry, the distribution of the *direction* of the Gaussian vector $g$ is actually uniform, and thus the probability that $f$ passes the test should be roughly equal to the probability that a uniformly random hyperplane through the origin separates the vectors

[18] The probabilistic expressions in this section are only valid for Boolean-valued $f$, but the other expressions make sense for general $f : \{-1, 1\}^n \to \mathbb{R}$, and we'll need these more general definitions in the next section.

$a$ and $b$, which, as we saw in the analysis of Goemans-Williamson, is

$$\frac{\arccos(a \cdot b)}{\pi} = \frac{\arccos\left(\sum_{i=1}^{n} a_i^2 \mu_i\right)}{\pi}$$

With high probability over $\mu$, we'll have (by a law of large numbers) that $\sum_{i=1}^{n} a_i^2 \mu_i \approx \rho$, and hence we don't expect a low-influence half-space function to pass the test with probability higher than $\mathrm{MAJ}_n$ (see (21)). We can now state the main ingredient needed for the analysis of the PCP:

**Majority is Stablest Theorem, version 2:** Let $\epsilon > 0$ and $\rho \in (-1, 0)$ be fixed. Then there exist constants $K, \delta > 0$ such that if $f : \{-1,1\}^n \to [-1,1]$ satisfies

$$\mathrm{Inf}_i^K(f) = \sum_{S \ni i, |S| \leq K} \hat{f}(S)^2 \leq \delta$$

for each coordinate $i$, then

$$S_\rho(f) = \sum_S \rho^{|S|} \hat{f}(S)^2 \geq 1 - \frac{2}{\pi} \arccos \rho - \epsilon$$

The proof, as given in [9], first establishes a general *invariance principle* for multilinear polynomials with low influence. An invariance principle is a type of central limit theorem that holds for a wider class of multilinear functions than just linear combinations as in the usual CLT. More precisely, in [9] the authors show that for functions of the form

$$Q(X_1, \ldots, X_n) = \sum_{S \subseteq [n]} c_S \prod_{i \in S} X_i$$

with small influences $\sum_{S \ni i} c_S^2 \leq \tau$ for all $i$, and independent random variables $X_i$ with mean 0, variance 1 and $\mathbb{E}|X_i|^3 \leq \beta$, we have

$$\sup_t |\Pr[Q(X_1, \ldots, X_n) \leq t] - \Pr[Q(G_1, \ldots, G_n) \leq t]| \leq O(d\beta^{1/3}\tau^{1/8d})$$

where the $G_i$ are independent standard Gaussians and $d$ is the degree of $Q$. We won't discuss the proof here[19], as the analysis is quite technical, and we don't want to keep the impatient reader waiting any longer. Let's check out the PCP.

### 3.4 The 2-query $\neq$ PCP of Khot, Kindler, Mossel and O'Donnell

Suppose we're given an instance $(V, W, E, [m], \{\pi_e\})$ of Label Cover, which is either at least $(1-\eta)$-satisfiable or at most $\gamma$-satisfiable, where we'll choose $\eta, \gamma$ to be sufficiently small later. Our verifier will expect as a proof the long code of the label of each vertex $w \in W$. Note that in Hastad's PCP, our verifier expected the proof to provide long codes for every vertex in $V$ as well as $W$. With only 2 queries, however, it's not clear how we could do anything useful

[19] When Khot et al. originally published [7], the full proof of Majority is Stablest was not known, and it was left as a conjecture which was later resolved by [9].

with knowledge of one $v$ label and one $w$ label, which is why our two queries will ask about two vertices $w, w' \in W$ that *share a neighbor* $v \in V$.

**The KKMO PCP:** Select a random vertex $v \in V$ uniformly at random, and two of its neighbors $w, w' \in W$ at random. Suppose $f_w$ and $f_{w'}$ are the supposed long codes of the labels for $w$ and $w'$ respectively, and let $\pi = \pi_{(v,w)}$ and $\pi' = \pi_{(v,w')}$. We then choose $x \in \{-1,1\}^m$ uniformly at random, and (like Hastad) choose a noisy $\mu \in \{-1,1\}^m$, with independent bits equal to 1 with probability $\frac{1}{2} + \frac{1}{2}\rho$, and -1 otherwise, where $\rho \in (-1, 0)$ will be chosen later. Then accept iff

$$f_w(x \circ \pi) \neq f_{w'}((x\mu) \circ \pi'). \tag{22}$$

**Completeness:** Fix a labelling which satisfies at least $(1-\eta)$-fraction of the edges. Then a simple union bound shows with probability at least $(1-2\eta)$, both of the edges $(v,w),(v,w')$ we select will be satisfied by this labelling. If $f_w$ and $f_{w'}$ are actually long codes of labels $a(w)$ and $a(w')$, then $f_w(x \circ \pi) = x_{\pi(a(w))} = x_{a(v)}$, while $f_{w'}(x\mu \circ \pi') = \mu_{a(v)} x_{\pi'(a(w'))} = \mu_{a(v)} x_{a(v)}$, which are unequal precisely when $\mu_{a(v)} = -1$, which happens with probability $\frac{1}{2} - \frac{1}{2}\rho$. Thus, the completeness of the test is at least $(1-2\eta)(\frac{1}{2} - \frac{1}{2}\rho)$.

**Soundness:** We show that if some proof causes the test to pass with probability at least $(\arccos\rho)/\pi + \epsilon$, then we can extract from it a labelling which satisfies at least a $\gamma = \gamma(\epsilon, \rho)$ fraction of the edge constraints. As this number will not depend on $m = |\Sigma|$, we can take $m$ large enough so that the UGC says GAP-ULC$_{1-\eta,\gamma}([m])$ is NP-hard. Using independence, we can derive the following formula for the acceptance probability:

$$\begin{aligned}
\Pr[acc] &= \mathbb{E}_{v,w,w',x,\mu}[\frac{1}{2} - \frac{1}{2} f_w(x \circ \pi) f_{w'}(x\mu \circ \pi')] \\
&= \frac{1}{2} - \frac{1}{2}\mathbb{E}_{v,x,\mu}\left[\mathbb{E}_w[f_w(x \circ \pi)]\mathbb{E}_{w'}[f_{w'}(x\mu \circ \pi')]\right]
\end{aligned} \tag{23}$$

If we define, for each $v \in V$,

$$g_v(x) = \mathbb{E}_{w \sim v}[f_w(x \circ \pi_w)]$$

then (23) becomes

$$\Pr[acc] = \frac{1}{2} - \frac{1}{2}\mathbb{E}_v[\mathbb{E}_{x,\mu}[g_v(x) g_v(x\mu)]] = \frac{1}{2} - \frac{1}{2}\mathbb{E}_v[S_\rho(g_v)]$$

Glancing back at our test (22), this formula actually seems quite natural – conversely, the test itself now seems a bit less mysterious, and it shouldn't be hard to see why the Majority is Stablest theorem is going to be useful here. If $\Pr[acc] > (\arccos\rho)/\pi + \epsilon$, then by an averaging argument,

$$S_\rho(g_v) \leq 1 - \frac{2}{\pi}\arccos\rho - \epsilon \tag{24}$$

for at least a fraction $\epsilon/2$ of the vertices $v \in V$ – call such vertices *good*. By the Majority is Stablest theorem, for good $v$, the function $g_v$ must have at least one influential coordinate, say $j_v$, such that $g_v$ has enough of its Fourier mass on the coefficients influenced by $j_v$ to make $g_v$ look (very roughly) like the dictator function $f(x) = x_{j_v}$. Since $g_v$ is an average of $f_w(x \circ \pi_w)$ over all neighboring $w$, it should be the case that a decent fraction of these $w$ look (very roughly) like the dictator function $f(x) = x_{\pi_w^{-1}(j_v)}$. Thus, if we label all of the vertices $v$

and $w$ by the index of the dictator functions they roughly resemble, we should have a decent chance of satisfying a decent fraction of the edge constraints. Let's carry out this argument more formally using basic Fourier analysis, in a way reminiscent of Hastad.

By our choice of label $j_v$ from the Majority is Stablest theorem,

$$\mathrm{Inf}^K_{j_v} g_v = \sum_{S \ni j_v, |S| \le K} \hat{g}_v(S)^2 \ge \delta$$

By linearity, one can check from the definition of $g_v$ that

$$\hat{g}_v(S) = \mathbb{E}_{w \sim v}[\hat{f}(\pi_w^{-1}(S))]$$

and hence

$$\delta \le \sum_{S \ni j_v, |S| \le K} \hat{g}_v(S)^2 = \sum_{S \ni j_v, |S| \le K} (\mathbb{E}_{w \sim v}[\hat{f}(\pi_w^{-1}(S))])^2 \le \sum_{S \ni j_v, |S| \le K} \mathbb{E}_{w \sim v}[\hat{f}(\pi_w^{-1}(S))^2] \quad (25)$$

Here it becomes important that the projections $\pi_w$ are bijections – this gives us the equality of sets

$$\{\pi_w^{-1}(S) : j_v \in S, |S| \le K\} = \{T : \pi_w^{-1}(j_v) \in T, |T| \le K\}$$

and thus the sum on the right in (25) is really $\mathbb{E}_{w \sim v}[\mathrm{Inf}_{\pi_w^{-1}(j_v)}(f_w)]$. By another averaging argument, at least a $\delta/2$ fraction of $w$ adjacent to each good $v$ have $\mathrm{Inf}_{\pi_w^{-1}(j_v)}(f_w) \ge \delta/2$. For any $w$, we'll choose a label uniformly at random from the set

$$C_w := \{i \in [m] : \mathrm{Inf}^K_i(f_w) \ge \delta/2\}$$

(or a just any random label if $C_w$ is empty.) Since $\sum_{i \in [m]} \mathrm{Inf}^K_i(f) = \sum_{|S| \le K} |S| \hat{f}(S)^2 \le K \sum_S \hat{f}(S)^2 = K$, it follows that $|C_w| \le 2K/\delta$. Thus, for a good edge $v$, at least a fraction $(\delta/2)(\delta/2K)$ of edges $(v, w)$ are satisfied in expectation. Since at least a fraction $\epsilon/2$ vertices $v$ are good, we've proven the existence of a labelling which satisfies a fraction

$$\gamma := (\epsilon/2)(\delta/2)(\delta/2K)$$

of edge constraints. Thus, we have soundness $\frac{\arccos \rho}{\pi} + \epsilon$.

Since $\eta$ and $\epsilon$ can be taken as small as we like, our PCP implies a hardness of approximation factor[20]

$$\frac{\arccos \rho}{\frac{1}{2} - \frac{1}{2}\rho} + \epsilon$$

for any $\rho \in (-1, 0)$ and $\epsilon > 0$: optimizing over $\rho$ in this interval recovers precisely the Goemans-Williams constant, $\alpha_{GW}$, plus an arbitrarily small $\epsilon$. This implies:

**Theorem:**[Khot, Kindler, Mossel, O'Donnell, 2005] *If we can $(\alpha_{GW} + \epsilon)$-approximate MAX-CUT in polynomial time for some $\epsilon > 0$, then for sufficiently small $\eta, \gamma > 0$, GAP-ULC$_{1-\eta,\gamma}([m])$ is in* P *for all $m$.*

[20] This factor is (up to $\epsilon + o(1)$ error) equal to the ratio of the probability that $\mathrm{MAJ}_n$ passes the $\neq$ test with parameter $\rho$ to the probability that a dictator passes the test.

## References


[1] Arora, S., Lund, C., Motawani, R., Sudan, M., and Szegedy, M. Proof Verification and the hardness of approximation problems. *Journal of the ACM,* 45(3): 505–555, 1998.

[2] Crescenzi, P., Silvestri, R., and Trevisan, L. On weighted vs unweighted versions of combinatorial optimization problems. *Information and Computation*, 167(1): 10–26, 2001.

[3] Hastad, J., Clique is hard to approximate within $n^{1-\epsilon}$. *Acta Math.,* 182:105–142, 1999.

[4] Hastad, J., Some optimal inapproximability results. *Journal of the ACM,* 48:798–869, 2001.

[5] Kahn, J., Kalai, G. and Linial, N. The influence of variables on boolean functions. In *Proc. 29th Ann. IEEE Symp. on Foundations of Comp. Sci.,* 68–80, 1988.

[6] Khot, S. On the Unique Games Conjecture, ftp.cs.nyu.edu/ khot/papers/UGCSurvey.pdf

[7] Khot, S., Kindler, G., Mossel, E. and O'Donnell, R. Optimal Inapproximability Results for MAX-CUT and Other 2-Variable CSPs? *Electronic Colloquium on Computational Complexity, Report No. 101,* 2005.

[8] Khot, S., Vishnoi, N. The unique games conjecture, integrality gap for cut problems and embeddability of negative type metrics into $\ell_1$. In *Proc. 46th IEEE Symposium on Foundations of Computer Science*, 2005.

[9] Mossel, E., O'Donnell, R., Oleszkiewicz, K. Noise stability of functions with low influences: Invariance and optimality. *Annals of Mathematics,* 171: 295 – 341, 2005.

[10] Raghavendra, P. Optimal algorithms and inapproximability results for every csp? In *Proc. ACM Symposium on the Theory of Computing*, 245–254, 2008.

[11] Raz, R. A parallel repetition theorem. *SIAM J. of Computing*, 27(3): 763-803, 1998.

[12] Trevisan, L., Sorkin, G., Sudan, M., and Williamson, D. Gadgets, approximation, and linear programming. *SIAM J. of Computing*, 29, 2074–2097.