Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 42 (2000)

The Non-Approximability of Non-Boolean Predicates

Lars Engebretsen*

Department of Numerical Analysis and Computer Science
Royal Institute of Technology
SE-100 44 Stockholm
SWEDEN
E-mail: enge@nada.kth.se

August 2001

Abstract. Constraint satisfaction programs where each constraint depends on a constant number of variables have the following property: The randomized algorithm that guesses an assignment uniformly at random satisfies an expected constant fraction of the constraints. By combining constructions from interactive proof systems with harmonic analysis over finite groups, Håstad showed that for several constraint satisfaction programs this naive algorithm is essentially the best possible unless $\mathbf{P} = \mathbf{NP}$. While most of the predicates analyzed by Håstad depend on a small number of variables, Samorodnitsky and Trevisan recently extended Håstad's result to predicates depending on an arbitrarily large, but still constant, number of Boolean variables.

We combine ideas from these two constructions and prove that there exists a large class of predicates on finite non-Boolean domains such that for predicates in the class, the naive randomized algorithm that guesses a solution uniformly is essentially the best possible unless $\mathbf{P} = \mathbf{NP}$. As a corollary, we show that the k-CSP problem over domains with size D cannot be approximated within $D^{k-O(\sqrt{k})} - \epsilon$, for any constant $\epsilon > 0$, unless $\mathbf{P} = \mathbf{NP}$. This lower bound matches well with the best known upper bound, D^{k-1}, of Serna, Trevisan and Xhafa.

1 Introduction

In a breakthrough paper, Håstad [6] studied the problem of giving approximate solutions to maximization versions of several constraint satisfaction problems. An instance of a such a problem is given as a collection of constraints, i.e., functions from some domain to $\{0, 1\}$, and the objective is to

*Part of this research was performed while the author was visiting MIT with support from the Marcus Wallenberg Foundation and the Royal Swedish Academy of Sciences.

ISSN 1433-8092

satisfy as many constraints as possible. An approximate solution of a constraint satisfaction program is simply an assignment that satisfies roughly as many constraints as possible. In this setting, we are interested in proving either that there exists a polynomial time algorithm producing approximate solutions some constant fraction from the optimum or that no such algorithms exist.

Typically, each individual constraint depends on a fixed number k of the variables and the size of the instance is given as the total number of variables that appear in the constraints. In this case, which is usually called the Max k-CSP problem, there exists a very naive algorithm that approximates the optimum within a constant factor: The algorithm that just guesses a solution at random. In his paper, Håstad [6] proved the very surprising fact that this algorithm is essentially the best possible efficient algorithm for several constraint satisfaction problems, unless $\mathbf{P} = \mathbf{NP}$. The proofs unify constructions from interactive proof systems with harmonic analysis over finite groups and give a general framework for proving strong impossibility results regarding the approximation of constraint satisfaction programs. Håstad [6] suggests that predicates with the property that the naive randomized algorithm is the best possible polynomial time approximation algorithm should be called *non-approximable beyond the random assignment threshold.*

Definition 1. *A Max k-CSP on k variables is* non-approximable beyond the random assignment threshold *if, for any constant $\epsilon > 0$, it is **NP**-hard to approximate the optimum of the CSP within a factor $w - \epsilon$, where $1/w$ is the expected fraction of constraints satisfied by a solution guessed uniformly at random.*

Håstad's paper [6] deals mainly with constraint satisfaction programs involving a small number of variables, typically three or four. In most of the cases, the variables are Boolean, but Håstad also treats the case of linear equations over Abelian groups. In the Boolean case, Håstad's techniques have been extended by Trevisan [13], Sudan and Trevisan [10], and Samorodnitsky and Trevisan [8] to some predicates involving a large, but still constant, number of Boolean variables. In this paper, we prove that those extensions can be adapted also to the non-Boolean case—a fact that is not immediately obvious from the proof for the Boolean case. This establishes non-approximability beyond the random assignment threshold for a large class of non-Boolean predicates. Our proofs use Fourier analysis of functions from finite Abelian groups to the complex numbers combined with what has now become standard constructions from the world of interactive proof systems. As a technical tool, Sudan and Trevisan [10] developed a certain composition lemma. In this paper, we extend this lemma to the non-Boolean setting. By using the lemma as an integrated part of the construction rather than a black box, we are also able to improve some of the constants involved.

A consequence of our result is that it is impossible to approximate Max k-CSP over domains of size D within $D^{k-O(\sqrt{k})} - \epsilon$, for any constant $\epsilon > 0$, in polynomial time unless $\mathbf{P} = \mathbf{NP}$. This lower bound matches well with the best known upper bound, D^{k-1}, following from a linear relaxation combined with randomized rounding [9, 12].

The paper is outlined as follows: We give the general ideas behind our construction in Sec. 2. Then we discuss Fourier transforms over Abelian groups and its connection to the so called *Long G-Code* in Secs. 3 and 4. We give the technical details of our construction in Sec. 5 and the connection to Max k-CSP and the non-approximability beyond the random assignment threshold of several non-Boolean predicates in Sec. 6. Finally, we conclude with some directions for future research.

2 Outline of the construction

The underlying idea in our construction is the same as in Håstad's [6]. We start with an instance of μ-gap E3-Sat(5).

Definition 2. μ-gap E3-Sat(5) *is the following decision problem: We are given a Boolean formula ϕ in conjunctive normal form, where each clause contains exactly three literals and each literal occurs exactly five times. We know that either ϕ is satisfiable or at most a fraction $\mu < 1$ of the clauses in ϕ are satisfiable and are supposed to decide if the formula is satisfiable.*

It is known [2, 3] that μ-gap E3-Sat(5) is $\mathbf{NP}$-hard.

There is a well-known two-prover one-round (2P1R) interactive proof system that can be applied to μ-gap E3-Sat(5). It consists of two provers, P_1 and P_2, and one verifier. Given an instance, i.e., an E3-Sat formula ϕ, the verifier picks a clause C and variable x in C uniformly at random from the instance and sends x to P_1 and C to P_2. It then receives an assignment to x from P_1 and an assignment to the variables in C from P_2, and accepts if these assignments are consistent and satisfy C. If the provers are honest, the verifier always accepts with probability 1 when ϕ is satisfiable, i.e., the proof system has *completeness* 1. It can be shown that the provers can fool the verifier with probability at most $(2 + \mu)/3$ when ϕ is not satisfiable, i.e., that the above proof system has *soundness* $(2 + \mu)/3$.

The soundness can be lowered to $((2+\mu)/3)^u$ by repeating the protocol u times independently, but it is also possible to construct a one-round proof system with lower soundness by repeating u times in parallel as follows: The verifier picks u clauses $(C_1, \ldots, C_u)$ uniformly at random from the instance. For each C_i, it also picks a variable x_i from C_i uniformly at random. The verifier then sends $(x_1, \ldots, x_u)$ to P_1 and the clauses $(C_1, \ldots, C_u)$ to P_2. It receives an assignment to $(x_1, \ldots, x_u)$ from P_1 and an assignment to the variables in $(C_1, \ldots, C_u)$ from P_2, and accepts if these assignments are consistent and satisfy $C_1 \wedge \cdots \wedge C_u$. As above, the completeness of this proof system is 1, and it can be shown [7] that the soundness is at most c_μ^u, where $c_\mu < 1$ is some constant depending on μ but not on u or the size of the instance.

In the above setting, the proof is simply an assignment to all the variables. In that case, the verifier can just compare the assignments it receives from the provers and check if they are consistent and satisfying. The construction we use to prove that several non-Boolean constraint satisfaction programs are non-approximable beyond the random assignment threshold can be viewed as a simulation of the u-parallel repetition of the above 2P1R interactive proof system for μ-gap E3-Sat(5). We use a probabilistically checkable proof system (PCP) with a verifier closely related to the particular constraint we want to analyze. To find predicates that depend on variables from some domain of size D and are non-approximable beyond the random assignment threshold, we work with an Abelian group G of size D. The predicates we study are ANDs of linear equations involving three variables in G. The proof is what Håstad [6] calls a Standard Written G-Proof with parameter u. It is supposed to be a very redundant encoding of a string of length n, which when ϕ is a satisfiable formula should be a satisfying assignment.

Definition 3. *If U is some set of variables taking values in $\{-1, 1\}$, we denote by $\{-1,1\}^U$ the set of every possible assignment to those variables. The* Long G-Code *of some string x of length $|U|$ is the value of all functions from $\{-1,1\}^U$ to G evaluated on the string x; $A_{U,x}(f) = f(x)$.*

Since there are $|G|^{2^{|U|}}$ functions from $\{-1,1\}^U$ to G, the *Long G-Code* of a string of length u has length $|G|^{2^u}$. The proof introduced by Håstad [6] contains the Long G-Code of several subsets containing a constant number of variables. Each such subset is supposed to represent either an assignment to the variables sent to P_1 or the clauses sent to P_2 in the 2P1R interactive proof system for μ-gap E3-Sat(5).

Definition 4. *A* Standard Written G-Proof with parameter u *contains for each set $U \subseteq [n]$ of size at most u a string of length $|G|^{2^{|U|}}$, which we interpret* *as the table of a function $A_U: \mathcal{F}_U^G \to G$. It also contains for each set W constructed as the set of variables in u clauses a function $A_W: \mathcal{F}_W^G \to G$.*

Definition 5. *A Standard Written G-Proof with parameter u is a* correct proof *for a formula ϕ of n variables if there is an assignment x, satisfying ϕ, such that A_V is the Long G-Code of $x|_V$ for any V of size at most u or any V constructed as the set of variables of u clauses.*

To check that the proof is a correct proof, we—following the construction of Samorodnitsky and Trevisan [8]—first query $2k$ positions from the proof and then, as a checking procedure, construct k^2 linear equations, each of them involving two of the first $2k$ queried positions and one extra variable. To give a more illustrative picture of the procedure, let the first $2k$ queries correspond to the vertices of a complete $k \times k$ bipartite graph. The k^2 linear equations that we check then correspond to the edges of this graph.

As for the non-approximability beyond the random assignment threshold, a random assignment to the variables satisfy all k^2 linear equations simultaneously with probability $|G|^{-k^2}$—the aim of our analysis is to prove that this is essentially the best possible any polynomial time algorithm can accomplish. This follows from the connection between our PCP and the 2P1R interactive proof system for μ-gap E3-Sat(5): We assume that it is possible to satisfy a fraction $|G|^{-k^2} + \epsilon$ for some constant $\epsilon > 0$ and prove that this implies that there is a correlation between the tables queried by the verifier in our PCP. We can then use this correlation to explicitly construct strategies for the provers in the 2P1R proof system for μ-gap E3-Sat(5) such that the verifier in that proof system accepts with probability larger than c_μ^u. The final link in the chain is the observation that since our verifier uses only logarithmic randomness, we can form a CSP with polynomial size by enumerating the checked constraints for every possible outcome of the random bits. If the resulting constraint satisfaction program is approximable beyond the random assignment threshold, we can use it to decide the **NP**-hard language μ-gap E3-Sat(5) in polynomial time.

We remark, that by checking the equations corresponding to some subset E of the edges in the complete bipartite graph we also get a predicate which is non-approximable beyond the random assignment threshold: It is satisfied with probability $|G|^{-|E|}$ by a random assignment and our proof methodology works also for this case.

3 Fourier Transforms

To prove a bound on the soundness of their verifiers Håstad [6], as well as Samorodnitsky and Trevisan [8], use Fourier transforms. In this section we give a brief account of the methods involved, for more details see Håstad's paper [6] or Terras's book [11].

Definition 6. *For a finite Abelian group G, the space $L^2(G)$ is the vector space of all functions from G to $\boldsymbol{C}$ equipped with the inner product*

$$\langle f_1, f_2 \rangle = \frac{1}{|G|} \sum_{g \in G} f_1(g)\overline{f_2(g)}. \tag{1}$$

The aim of Fourier transforms is to express functions as linear combinations of basis functions with certain nice properties. To define the basis functions for the space $L^2(G)$, we use the fact that every finite Abelian group G, can be represented as $G \cong \boldsymbol{Z}_{i_1} \times \cdots \times \boldsymbol{Z}_{i_k}$, where $|G| = i_1 \cdots i_k$. An element $g \in G$ is represented as a k-tuple

$$g \sim (g^{(1)}, \ldots, g^{(k)}) \in \boldsymbol{Z}_{i_1} \times \cdots \times \boldsymbol{Z}_{i_k}.$$

We use multiplication as the group operation and 1 as the group identity. If two group elements g_1 and g_2 have the representations

$$\begin{aligned} g_1 &\sim (g_1^{(1)}, \ldots, g_1^{(k)}), \\ g_2 &\sim (g_2^{(1)}, \ldots, g_2^{(k)}), \end{aligned}$$

respectively, the element $g_1 g_2$ has the representation

$$g_1 g_2 \sim (g_1^{(1)} + g_2^{(1)} \bmod i_1, \ldots, g_1^{(k)} + g_2^{(k)} \bmod i_k).$$

The group identity is represented by a vector of k zeros.

Definition 7. *The set $\boldsymbol{T}$ is the set of all complex numbers of unit norm. The set of* characters *of an Abelian group G is the set of all linear homomorphisms from G to $\boldsymbol{T}$.*

For an Abelian group G, the characters are

$$\psi_a(g) = \prod_{j=1}^{k} \exp\left(\frac{2\pi i a^{(j)} g^{(j)}}{i_j}\right). \tag{2}$$

They are homomorphisms from G to $\boldsymbol{T}$ since

$$\psi_a(g_1 g_2) = \psi_a(g_1)\psi_a(g_2). \tag{3}$$

Below, we need also the following identities involving the characters of G:

$$\psi_{a_1 a_2}(g) = \psi_{a_1}(g)\psi_{a_2}(g), \tag{4}$$

$$\psi_{1_G}(g) = 1, \tag{5}$$

$$\sum_{a\in G} \psi_a(g) = \begin{cases} |G| & \text{if } g = 1, \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

In fact, the characters of G form an orthonormal basis for $L^2(G)$. This follows from a completeness argument: The characters are many enough, and they are orthogonal since if $a \neq a'$,

$$\langle \psi_a, \psi_{a'} \rangle = \frac{1}{|G|} \sum_{g\in G} \psi_a(g)\overline{\psi_{a'}(g)} = \frac{1}{|G|} \sum_{g\in G} \psi_a(gg')\overline{\psi_{a'}(gg')}$$

for some arbitrary $g' \in G$. The last equality holds since we sum over all elements in G. By the homomorphism property (3), we can rewrite the last expression as

$$\frac{1}{|G|} \sum_{g\in G} \psi_a(g)\psi_a(g')\overline{\psi_{a'}(g)\psi_{a'}(g')} = \frac{\psi_a(g')\overline{\psi_{a'}(g')}}{|G|} \sum_{g\in G} \psi_a(g)\overline{\psi_{a'}(g)}.$$

If we choose g' such that $\psi_a(g') \neq \psi_{a'}(g')$, this is always possible since $a \neq a'$, we have shown that $\langle \psi_a, \psi_{a'} \rangle = c\langle \psi_a, \psi_{a'} \rangle$ for some $c \neq 1$. Thus, $\langle \psi_a, \psi_{a'} \rangle = 0$ if $a \neq a'$. Finally, if $a = a'$,

$$\langle \psi_a, \psi_a \rangle = \frac{1}{|G|} \sum_{g\in G} \psi_a(g)\overline{\psi_a(g)} = \frac{1}{|G|} \sum_{g\in G} |\psi_a(g)|^2 = 1$$

since $\psi_a(g)$ has unit norm. Thus the characters are orthonormal.

Once we have our orthonormal basis, the definition of the Fourier transform of a function in $L^2(G)$ straightforward.

Definition 8. *For a finite Abelian group G, the Fourier coefficients $\{\hat{f}_a\}_{a\in G}$ of a function in $L^2(G)$ are $\hat{f}_a = \langle f, \psi_a \rangle$, where $\{\psi_a\}_{a\in G}$ are the characters of G.*

As an illustration of these concepts, we state and prove the only theorem from classical Fourier analysis that we use in this paper, namely *Parseval's equality.* Using only the concept of orthonormality, the theorem provides a relationship between a function and its Fourier coefficients.

Theorem 1. *Suppose that f is a function in $L^2(G)$ and that its Fourier coefficients are $\{\hat{f}_a\}_{a\in G}$. Then $\langle f, f \rangle = \sum_{a\in G} |\hat{f}_a|^2$.*

Proof. If we expand f in its Fourier series, we get

$$\langle f, f \rangle = \sum_{a\in G} \sum_{a'\in G} \hat{f}_a \overline{\hat{f}_{a'}} \langle \phi_a, \phi_{a'} \rangle = \sum_{a\in G} \hat{f}_a \overline{\hat{f}_a} = \sum_{a\in G} |\hat{f}_a|^2,$$

where the second equality follows since the characters are orthonormal. ∎

4 The Long G-Code

To prove our lower bound, we need to use the Fourier transform of the *Long G-Code*. The techniques in this section were pioneered by Håstad [6]; Terras [11] surveys and analyzes several other applications of the Fourier transform. What makes the Fourier transform extremely useful in combination with the Long G-Code seems to be that the characters of G can be used both to form a Fourier basis of functions from the Long G-Code to $\boldsymbol{C}$ and to form certain predicates needed in the analysis of certain tests on codewords.

Definition 9. *If U is some set of variables taking values in $\{-1, 1\}$, we denote by $\{-1, 1\}^U$ the set of every possible assignment to those variables. Define*

$$\mathcal{F}_U^G = \{f : \{-1, 1\}^U \to G\}. \tag{7}$$

For a function $f \in \mathcal{F}_U^G$, we denote by $|f|$ the number of x such that $f(x) \neq 1_G$.

Definition 10. *The space $L^2(\mathcal{F}_U^G)$ is the vector space of all functions from $\mathcal{F}_U^G$ to $\boldsymbol{C}$ equipped with the inner product*

$$\langle F_1, F_2 \rangle = \frac{1}{|G|^{2^{|U|}}} \sum_{f \in \mathcal{F}_U^G} F_1(f)\overline{F_2(f)}. \tag{8}$$

To shorten the notation, we frequently write the above expression as

$$\langle F_1, F_2 \rangle = \mathrm{E}_{f \in \mathcal{F}_U^G}[F_1(f)\overline{F_2(f)}], \tag{9}$$

where it is understood that the probability distribution involved is the uniform distribution.

We can view the Long G-Code as a function $A : \mathcal{F}_U^G \to G$ From now on, it is understood that this function corresponds to some string $x \in \{-1, 1\}^U$, which is interpreted as assignments to the variables in U.

In Section 5 we want to estimate the probability that certain tests over the group G accept. It turns out that an important technical tool in these efforts is the Fourier transform on functions in the space $L^2(\mathcal{F}_U^G)$. To obtain our Fourier basis, we need an expression for the characters of $\mathcal{F}_U^G$. To derive that expression, we note that we can identify this space with $L^2(G^{2^{|U|}})$ by identifying a function f with a table of the values $f(x)$ for every $x \in \{-1, 1\}^U$. Thus, the characters of $\mathcal{F}_U^G$ are

$$\chi_\alpha(f) = \prod_{x \in \{-1,1\}^U} \psi_{\alpha(x)}(f(x)), \tag{10}$$

where $\psi_a(g)$ is the corresponding group character of G. In this definition, α is a function from $\{-1,1\}^U$ to G. In the same way as the characters of G, the characters of $\mathcal{F}_U^G$ satisfy the following identities:

$$\chi_\alpha(f_1 f_2) = \chi_\alpha(f_1)\chi_\alpha(f_2), \tag{11}$$

$$\chi_{\alpha_1\alpha_2}(f) = \chi_{\alpha_1}(f)\chi_{\alpha_2}(f), \tag{12}$$

$$\chi_1(f) = 1, \tag{13}$$

$$\mathrm{E}_{f\in\mathcal{F}_U^G}[\chi_\alpha(f)] = \begin{cases} 1 & \text{if } \alpha = 1, \\ 0 & \text{otherwise.} \end{cases} \tag{14}$$

We can now define the Fourier coefficients as usual, $\hat{F}_\alpha = \langle F, \chi_\alpha \rangle$, for some function $F \in L^2(\mathcal{F}_U^G)$. This function then has the Fourier expansion

$$F = \sum_{\alpha\in\mathcal{F}_U^G} \hat{F}_\alpha \chi_\alpha. \tag{15}$$

In Section 5, we also need some technical lemmas regarding the Fourier transform of the Long G-Code in various settings.

4.1 A Projection Lemma

Suppose that $U \subseteq W$ and that $y \in \{-1,1\}^W$. Since y gives an assignment to all variables in W, we can use y to form an assignments to all variables in U.

Definition 11. *Let $U \subseteq W$ and $y \in \{-1,1\}^W$. Form $y|_U \in \{-1,1\}^U$ as follows: For every variable in U, choose the assignment prescribed by y.*

Definition 12. *Let $U \subseteq W$ and $\beta \in \mathcal{F}_W^G$. Form $\pi_U(\beta) \in \mathcal{F}_U^G$ as follows:*

$$(\pi_U(\beta))(x) = \prod_{y:y|_U=x} \beta(y). \tag{16}$$

Using the projection equality (16) and the fact that G is Abelian, we see that $\pi_U(\beta^{-1}) = (\pi_U(\beta))^{-1}$.

Lemma 1. *Let $U \subseteq W$ and let $\beta \in \mathcal{F}_W^G$ be arbitrary. Let $f \in \mathcal{F}_U^G$ be some arbitrary function and define a function $g \in \mathcal{F}_W^G$ such that $g(y) = f(y|_U)$. Then $\chi_\beta(g) = \chi_{\pi_U(\beta)}(f)$.*

Proof. By the definition of χ,

$$\chi_\beta(g) = \prod_{y\in\{-1,1\}^W} \psi_{\beta(y)}(g(y)).$$

Let us study the partition of $\{-1,1\}^W$ into sets of the form $\{y : y|_U = x\}$. On those sets, $g(y) = f(x)$, which means that we can write

$$\chi_\beta(g) = \prod_{x\in\{-1,1\}^U} \prod_{y:y|_U=x} \psi_{\beta(y)}(f(x)).$$

Since ψ_a is linear in a,

$$\prod_{y:y|_U=x} \psi_{\beta(y)}(f(x)) = \psi_{(\pi_U(\beta))(x)}(f(x)),$$

and thus

$$\chi_\beta(g) = \prod_{x\in\{-1,1\}^U} \psi_{(\pi_U(\beta))(x)}(f(x)) = \chi_{\pi_U(\beta)}(f),$$

where $\pi_U(\beta)$ is defined as in Definition 12. ∎

Note that the above lemma proves a relation between χ_β, which is a character of $L^2(\mathcal{F}_W^G)$, and $\chi_{\pi_U(\beta)}$, which is a character of $L^2(\mathcal{F}_U^G)$.

4.2 A Folding Lemma

Definition 13. *We say that a function $A\colon \mathcal{F}_U^G \to G$ is* folded over G *if*

$$A(\Gamma f) = \Gamma A(f) \tag{17}$$

for all $\Gamma \in G$ and all $f \in \mathcal{F}_U^G$.

Lemma 2. *Suppose that the function $A\colon \mathcal{F}_U^G \to G$ is folded over G. Let $\hat{A}_{\alpha,\gamma}$ be the Fourier coefficients of the function $\psi_\gamma \circ A$ for some $\gamma \in G$. Then $\hat{A}_{\alpha,\gamma} = 0$ unless*

$$\prod_{x\in\{-1,1\}^U} \alpha(x) = \gamma.$$

Proof. Since the expectation in the definition of the inner product is taken over all functions in $\mathcal{F}_U^G$ we can write $\hat{A}_{\alpha,\gamma}$ as

$$\hat{A}_{\alpha,\gamma} = \langle \psi_\gamma \circ A, \chi_\alpha \rangle = \mathrm{E}_{f \in \mathcal{F}_U^G}[\psi_\gamma(A(\Gamma f))\overline{\chi_\alpha(\Gamma f)}]$$

for any $\Gamma \in G$. By the folding equality (17) and the homomorphism property (3),

$$\psi_\gamma(A(\Gamma f)) = \psi_\gamma(\Gamma A(f)) = \psi_\gamma(\Gamma)\psi_\gamma(A(f)),$$

and by the homomorphism property (11),

$$\chi_\alpha(\Gamma f) = \chi_\alpha(\Gamma)\chi_\alpha(f).$$

Thus,

$$\langle \psi_\gamma \circ A, \chi_\alpha(f) \rangle = \psi_\gamma(\Gamma)\overline{\chi_\alpha(\Gamma)}\langle \psi_\gamma \circ A, \chi_\alpha(f) \rangle$$

for any $\Gamma \in G$, which means that $\psi_\gamma(\Gamma)\overline{\chi_\alpha(\Gamma)} = 1$ for all $\Gamma \in G$ if $\hat{A}_{\alpha,\gamma} \neq 0$. Since $\chi_\alpha(\Gamma)$ has unit norm, $\overline{\chi_\alpha(\Gamma)} = (\chi_\alpha(\Gamma))^{-1}$, and thus

$$\psi_\gamma(\Gamma) = \chi_\alpha(\Gamma) = \prod_{x \in \{-1,1\}^U} \psi_{\alpha(x)}(\Gamma)$$

for all $\Gamma \in G$ if $\hat{A}_{\alpha,\gamma} \neq 0$. Since ψ_a is linear in a,

$$\prod_{x \in \{-1,1\}^U} \psi_{\alpha(x)}(\Gamma) = \psi_a(\Gamma)$$

where $a = \prod_{x \in \{-1,1\}^U} \alpha(x)$. Thus $\psi_\gamma(\Gamma) = \psi_a(\Gamma)$ for all $\Gamma \in G$ if $\hat{A}_{\alpha,\gamma} \neq 0$. This can be true only if

$$\gamma = a = \prod_{x \in \{-1,1\}^U} \alpha(x),$$

which completes the proof. ∎

Corollary 1. *Suppose that the function $B\colon \mathcal{F}_W^G \to G$, where $U \subseteq W$, is folded over G. Let $\hat{B}_{\beta,\gamma}$ be the Fourier coefficients of the function $\psi_\gamma \circ B$ for some $\gamma \in G \setminus \{1_G\}$. Then, for all β such that $\hat{B}_\beta \neq 0$ there exists an $x \in \{-1,1\}^U$ such that $(\pi_U(\beta))(x) \neq 1_G$.*

Proof. Since $\hat{B}_{\beta,\gamma} \neq 0$, Lemma 2 implies that $\prod_{y\in\{-1,1\}^W} \beta(y) = \gamma$. We now express this product in G as

$$\prod_{x\in\{-1,1\}^U} \prod_{y\in\{-1,1\}^W : y|_U = x} \beta(y) = \gamma$$

and use the definition of the function $\pi_U(\beta)$ to obtain

$$\prod_{x\in\{-1,1\}^U} (\pi_U(\beta))(x) = \gamma.$$

Since $\gamma \neq 1_G$, this implies that there exists at least one $x \in \{-1,1\}^U$ such that $(\pi_U(\beta))(x) \neq 1_G$. ∎

4.3 A Conditioning Lemma

If f is a function in $\mathcal{F}_U^G$ and h is some Boolean function on $\{-1,1\}^U$, we define the function $f \wedge h$ as

$$(f \wedge h)(x) = \begin{cases} f(x) & \text{if } h(x) = \text{True}, \\ 1_G & \text{otherwise.} \end{cases} \tag{18}$$

Definition 14. *The function* $A\colon \mathcal{F}_U^G \to G$ *is* conditioned upon h *if* $A(f) = A(f \wedge h)$.

Lemma 3. *Suppose that the function* $A\colon \mathcal{F}_U^G \to G$ *is conditioned upon* h*. Let* $\hat{A}_{\alpha,\gamma}$ *be the Fourier coefficients of the function* $\psi_\gamma \circ A$ *for some* $\gamma \in G \setminus \{1_G\}$*. Then* $\hat{A}_{\alpha,\gamma} = 0$ *for any* α *such that there exists an* x *with the property that* $\alpha(x) \neq 1_G$ *and* $h(x) = \text{False}$.

Proof. Suppose that there exists an x_0 such that $\alpha(x_0) \neq 1_G$ and $h(x_0) =$ False. Write

$$\hat{A}_{\alpha,\gamma} = \langle \psi_\gamma \circ A, \chi_\alpha \rangle = \frac{1}{|G|^{2^{|U|}}} \sum_{\substack{f\in\mathcal{F}_U^G \\ f(x_0)=1}} \sum_{a\in G} \psi_\gamma(A(f_a))\overline{\chi_\alpha(f_a)}, \tag{19}$$

where f_a is defined from f as

$$f_a(x) = \begin{cases} a & \text{if } x = x_0, \\ f(x) & \text{otherwise.} \end{cases} \tag{20}$$

Since A is conditioned upon h and $h(x_0) =$ False, we can rewrite the expression (19) for $\hat{A}_{\alpha,\gamma}$ as

$$\hat{A}_{\alpha,\gamma} = \frac{1}{|G|^{2^{|U|}}} \sum_{\substack{f\in\mathcal{F}_U^G \\ f(x_0)=1}} \psi_\gamma(A(f)) \sum_{a\in G} \overline{\chi_\alpha(f_a)}.$$

By the definition of f_a,

$$\sum_{a \in G} \chi_\alpha(f_i) = \sum_{a \in G} \prod_{x \in \{-1,1\}^U} \psi_{\alpha(x)}(f_i(x))$$
$$= \prod_{\substack{x \in \{-1,1\}^U \\ x \neq x_0}} \psi_{\alpha(x)}(f(x)) \sum_{a \in G} \psi_{\alpha(x_0)}(a).$$

By the definition of inner product in $L^2(G)$,

$$\sum_{a \in G} \psi_{\alpha(x_0)}(a) = |G| \langle \psi_{\alpha(x_0)}, \psi_{1_G} \rangle.$$

Since $\alpha(x_0) \neq 1_G$ and the functions $\{\psi_a\}_{a \in G}$ are orthogonal in $L^2(G)$,

$$\langle \psi_{\alpha(x_0)}, \psi_1 \rangle = 0.$$

We conclude that $\hat{A}_\alpha = 0$ for all α such that there exists an x_0 such that $\alpha(x_0) \neq 1_G$ and $h(x_0) = \text{False}$. ∎

5 The Proof of the Lower Bound

The construction we use is essentially a reduction from μ-gap E3-Sat(5). We adapt the PCP construction of Samorodnitsky and Trevisan [8] to give a PCP with an acceptance predicate that is a function of roughly $k^2 + 2k$ variables in G. Then we prove that if the soundness of our PCP is high, we can decide μ-gap E3-Sat(5). On a slightly more detailed—but still high—level, the construction consists of the following steps:

1. Establish that there exists a two-prover one-round interactive proof system for μ-gap E3-Sat(5) with the following properties:
   (a) The queries to the provers and the answers from the provers have constant length.
   (b) The protocol has perfect completeness and soundness c_μ^u, where u is essentially the size of the queries to the provers.
2. Construct a PCP as follows:
   (a) The proof contains encodes answers to all possible queries in the above proof system for μ-gap E3-Sat(5).
   (b) The verifier, parameterized by the arbitrary constant $\delta_1 > 0$, accepts if a cleverly chosen constraint over G is satisfied.

The verifier uses certain conventions when accessing the proof. These conventions imply that certain bad proofs are accepted only with a small probability, and that Step 4 below is possible.

3. Assume that the verifier accepts an incorrect proof with probability $1/|G|+\delta_2$, where $\delta_2 > 0$ is some arbitrary constant, and prove that this implies that the tables in the proof are correlated. This correlation can be quantified by bounding a certain expression, involving the Fourier coefficients for some of the tables in the proof, by a function of δ_2.

4. Use the correlation between the tables to design a randomized strategy for the provers in the interactive proof system for μ-gap E3-Sat(5). Prove that if the provers follow this strategy, the verifier in the interactive proof system for μ-gap E3-Sat(5) accepts with probability greater than some function of δ_1 and δ_2. Conclude that the soundness of the interactive proof system for μ-gap E3-Sat(5) is at least c_{δ_1,δ_2}, which does not depend on u.

5. Choose the constant u in Step 1 such that $c_\mu^u < c_{\delta_1,\delta_2}$ and conclude that we have arrived at a contradiction.

We now do the above steps in detail.

5.1 An Interactive Proof System for μ-gap E3-Sat(5)

There is a well-known two-prover one-round interactive proof system that can be applied to μ-gap E3-Sat(5). It consists of two provers, P_1 and P_2, and one verifier. Given an instance, i.e., an E3-Sat formula ϕ, the verifier picks a clause C and variable x in C uniformly at random from the instance and sends x to P_1 and C to P_2. It then receives an assignment to x from P_1 and an assignment to the variables in C from P_2, and accepts if these assignments are consistent and satisfy C. If the provers are honest, the verifier always accepts with probability 1 when ϕ is satisfiable.

Lemma 4. *There exists provers that make the verifier accept a satisfiable instance of μ-gap E3-Sat(5) with probability* 1.

Proof. Let π be an assignment satisfying the instance and let both provers answer according to this assignment. ∎

Lemma 5. *The provers can fool the verifier to accept an unsatisfiable instance of μ-gap E3-Sat(5) with probability at most* $(2+\mu)/3$.

Proof. The strategy of P_1 defines an assignment π to all variables in the instance. Since the provers coordinate their strategies, we can assume that this assignment is known to P_2. Given this assignment, it is optimal for P_2 to proceed as follows: If it obtains a clause satisfied by π, it answers according to π. If it obtains a clause not satisfied by π it must answer with an assignment satisfying the clause, since verifier accepts if the assignment returned by P_2 satisfies the clause and is consistent with the assignment returned by P_1. Given a clause that is not satisfied by π, the probability that the verifier accepts is maximized if P_2 answers according to π for two of the three variables and inverts the answer of one variable. The variable P_2 inverts is chosen uniformly at random. Then the verifier accepts with probability $2/3$.

To sum up the above discussion, the provers can always fool the verifier when the verifier happens to choose a clause satisfied by π, and the fool the verifier with probability $2/3$ when the verifier happens to choose a clause not satisfied by π. If we let p denote the fraction of clauses satisfied by π, the verifier accepts with probability $p + (1-p)\frac{2}{3} = (2+p)/3$.

Finally, we note that we always have $p \leq \mu$, by the definition of μ-gap E3-Sat(5). This implies that the provers can make the verifier accept an unsatisfiable instance with probability at most $(2+\mu)/3$. ∎

To summarize the above analysis in the language of PCPs, the above proof system has completeness 1 and soundness $(2+\mu)/3$.

The soundness can be lowered to $((2+\mu)/3)^u$ by repeating the protocol u times independently, but it is also possible to construct a one-round proof system with lower soundness as follows: The verifier picks u clauses $(C_1, \dots, C_u)$ uniformly at random from the instance. For each C_i, it also picks a variable x_i from C_i uniformly at random. The verifier then sends $(x_1, \dots, x_u)$ to P_1 and the clauses $(C_1, \dots, C_u)$ to P_2. It receives an assignment to $(x_1, \dots, x_u)$ from P_1 and an assignment to the variables in $(C_1, \dots, C_u)$ from P_2, and accepts if these assignments are consistent and satisfy $C_1 \wedge \dots \wedge C_u$. As above, the completeness of this proof system is 1, and it can be shown [7] that the soundness is at most c_μ^u, where $c_\mu < 1$ is some constant depending on μ but not on u or the size of the instance.

5.2 The PCP

The proof is a Standard Written G-Proof with parameter u. It is supposed to represent a string of length n. When ϕ is a satisfiable formula this string should be a satisfying assignment. In Sec. 4.2 we saw that if a function from

$\mathcal{F}_U^G$ to G is folded over G, many of its Fourier coefficients vanish. It turns out that we need to have the above tables folded over G in order for the proof of the lower bound to work. This is not a problem, since the folding property can easily be enforced by the verifier as follows: When the verifier is supposed to query some position $A_U(f)$ from the proof, it instead queries $A_U(\Gamma^{-1}f)$, where $\Gamma \in G$ is chosen according to a fixed convention. Then the verifier uses the value $\Gamma A_U(\Gamma^{-1}f)$ as $A_U(f)$. An analogous procedure is used for the table representing A_W

The verifier is parameterized by the integers ℓ and m, a set $E \subseteq [\ell] \times [m]$, and a constant $\delta_1 > 0$; and it should accept with high probability if the proof is a correct Standard Written G-Proof for a given formula ϕ.

1. Select uniformly at random u variables $x_1, \ldots, x_u$. Let U be the set of those variables.
2. For $j = 1, \ldots, m$, select uniformly at random u clauses $C_{j,1}, \ldots, C_{j,u}$ such that clause $C_{j,i}$ contains variable x_i. Let Φ_j be the Boolean formula $C_{j,1} \wedge \cdots \wedge C_{j,u}$. Let W_j be the set of variables in the clauses $C_{j,1}, \ldots, C_{j,u}$.
3. For $i = 1, \ldots, \ell$, select uniformly at random $f_i \in \mathcal{F}_U^G$.
4. For $j = 1, \ldots, m$, select uniformly at random $g_j \in \mathcal{F}_{W_j}^G$.
5. For all $(i,j) \in E$, choose $e_{ij} \in \mathcal{F}_{W_j}^G$ such that, independently for all $y \in W$,
   (a) With probability $1 - \delta_1$, $e_{ij}(y) = 1_G$.
   (b) With probability δ_1, $e_{ij}(y)$ is selected uniformly at random from G.
6. Define h_{ij} such that $h_{ij}(y) = (f_i(y|_U)g_j(y)e_{ij}(y))^{-1}$.
7. If for all $(i,j) \in E$, $A_U(f_i)A_{W_j}(g_j \wedge \Phi_j)A_{W_j}(h_{ij} \wedge \Phi_j) = 1$, then accept, else reject.

Lemma 6. *The completeness of the above test is at least* $(1-\delta_1)^{|E|}$.

Proof. Given a correct proof, the verifier can only reject if one of the error functions e_{ij} are not 1_G for the particular string encoded in the proof. Since the error functions are chosen pointwise uniformly at random, the probability that they all evaluate to 1_G for the string encoded in the proof is $(1-\delta_1)^{|E|}$. Thus, the verifier accepts a correct proof with probability at least $(1-\delta_1)^{|E|}$. ∎

5.3 Expressing the Acceptance Probability

To shorten the notation, we define the shorthands $A(f) = A_U(f)$ and $B_j(g) = A_{W_j}(g \wedge \Phi_j)$.

Lemma 7. *The test in the PCP accepts with probability*

$$\frac{1}{|G|^{|E|}} \sum_{S \subseteq E} \mathrm{E}[T_S], \tag{21}$$

where

$$T_S = \prod_{(i,j) \in S} \Bigg(\sum_{\gamma \in G \setminus \{1\}} \psi_\gamma(A(f_i) B_j(g_j) B_j(h_{ij})) \Bigg). \tag{22}$$

We use the convention that $T_\emptyset = 1$.

Proof. The PCP tests if $|E|$ linear equations of the form

$$A(f_i) B_j(g_j) B_j(h_{ij}) = 1$$

over the group G are satisfied. We index the equations by (i,j), and note that the fact (5) that $\psi_{1_G}(g) = 1$ and the summation relation (6) together imply that the expression

$$P_{ij} = \frac{1}{|G|} \Bigg(1 + \sum_{\gamma \in G \setminus \{1\}} \psi_\gamma(A(f_i) B_j(g_j) B_j(h_{ij})) \Bigg)$$

is one when the equation corresponding to (i,j) is satisfied and zero otherwise. Since the test accepts if all equations are satisfied,

$$P = \prod_{(i,j) \in E} P_{ij} = \begin{cases} 1 & \text{if the test in the PCP accepts,} \\ 0 & \text{otherwise.} \end{cases}$$

Since the equations are chosen at random, P is an indicator random variable and we can write

$$\Pr[\text{The PCP accepts}] = \mathrm{E}[P].$$

If we expand the product in the definition of P, we arrive at the expression in (21) and (22). ∎

5.4 Identifying a Large Term

Lemma 8. *If the probability that the above test accepts is* $|G|^{-|E|} + \delta_2$ *for some* $\delta_2 > 0$*, then* $|\operatorname{E}[T_S]| \geq \delta_2$ *for some* $S \neq \emptyset$ *such that* $S \subseteq E$.

Proof. Suppose that $|\operatorname{E}[T_S]| < \delta_2$ for all $S \neq \emptyset$ such that $S \subseteq E$. Then

$$\Pr[\text{accept}] \leq \frac{1}{|G|^{|E|}} \sum_{S \subseteq E} |\operatorname{E}[T_S]| < \frac{1 + \delta_2(|G|^{|E|} - 1)}{|G|^{|E|}} < |G|^{-|E|} + \delta_2,$$

which is a contradiction. ∎

5.5 Bounding the Large Term

Lemma 9. *Suppose that* $|\operatorname{E}[T_S]| \geq \delta_2 > 0$ *for some set* $S \neq \emptyset$ *such that* $S \subseteq E$*. Number the vertices in this set* S *in such a way that there is at least one edge of the form* $(1, j)$ *and all edges of that form are* $(1, 1), \ldots, (1, d)$*. Let*

$$Q = \sum_{\substack{\alpha, \beta_1, \ldots, \beta_d \\ \alpha = \pi_U(\beta_1) \cdots \pi_U(\beta_d)}} |\hat{A}_\alpha|^2 |\hat{B}_{1,\beta_1}|^2 \cdots |\hat{B}_{d,\beta_d}|^2 (1 - \delta_1)^{2(|\beta_1| + \cdots + |\beta_d|)}, \tag{23}$$

where

$$W_j = \{\textit{variables in } \Phi_j\}, \tag{24}$$
$$A(f) = A_U(f), \tag{25}$$
$$B_j(g) = B_{W_j}(g \wedge \Phi_j), \tag{26}$$
$$\hat{A}_\alpha = \langle \psi_{\gamma_1 \cdots \gamma_d} \circ A, \chi_\alpha \rangle, \tag{27}$$
$$\hat{B}_{j,\beta_j} = \langle \psi_{\gamma_j} \circ B_j, \chi_{\beta_j} \rangle. \tag{28}$$

Then there exists $\gamma_1, \ldots, \gamma_d \in G \setminus \{1\}$ *such that*

$$\operatorname{E}_{U, \Phi_1, \ldots, \Phi_d}[Q] \geq \delta_2^2 / (|G| - 1)^{|S|}. \tag{29}$$

Proof. We split the product in the definition of T_S into the two factors

$$C_1 = \prod_{(i,j) \in S, i \neq 1} \Bigg(\sum_{\gamma \in G \setminus \{1_G\}} \psi_\gamma(A(f_i) B_j(g_j) B_j(h_{ij})) \Bigg)$$

and

$$C_2 = \prod_{j=1}^{d} \Bigg(\sum_{\gamma \in G \setminus \{1_G\}} \psi_\gamma(A(f_1) B_j(g_j) B_j(h_{1,j})) \Bigg).$$

Since C_1 is independent of f_1 and $e_{1,1}, \ldots, e_{1,k}$, we use conditional expectation to rewrite $\mathrm{E}[T_S]$. If we let $\mathrm{E}_1[\cdot]$ denote the expected value taken over the random variables f_1 and $e_{1,1}, \ldots, e_{1,k}$, we obtain

$$\mathrm{E}[T_S] = \mathrm{E}[\mathrm{E}_1[T_S]] = \mathrm{E}[C_1\, \mathrm{E}_1[C_2]].$$

This implies that

$$|\mathrm{E}[T_S]|^2 \leq \mathrm{E}\Big[|C_1|^2 |\mathrm{E}_1[C_2]|^2\Big].$$

By expanding the product in the definition of C_1, we obtain

$$|C_1|^2 = \left| \sum_{\gamma_1 \in G \setminus \{1_G\}} \cdots \sum_{\gamma_{|S|-d} \in G \setminus \{1_G\}} \prod_{j=1}^{|S|-d} \psi_{\gamma_j}(\cdot) \right|^2,$$

where we have suppressed the argument to ψ_{γ_j}. Since $\psi_{\gamma_j}(\cdot)$ is a complex root of unity, and a product of roots of unity is also a root of unity,

$$|C_1|^2 \leq \sum_{\gamma_1 \in G \setminus \{1_G\}} \cdots \sum_{\gamma_{|S|-d} \in G \setminus \{1_G\}} \left| \prod_{j=1}^{|S|-d} \psi_{\gamma_j}(\cdot) \right|^2 = (|G|-1)^{|S|-d}.$$

Thus,

$$|\mathrm{E}[T_S]|^2 \leq (|G|-1)^{|S|-d}\, \mathrm{E}\Big[|\mathrm{E}_1[C_2]|^2\Big].$$

Now we expand the product in the definition of C_2,

$$C_2 = \sum_{\gamma_1 \in G \setminus \{1_G\}} \cdots \sum_{\gamma_d \in G \setminus \{1_G\}} \prod_{j=1}^{d} \psi_{\gamma_j}(A(f_1) B_j(g_j) B_j(h_{1,j})).$$

If we write

$$C_3 = \prod_{j=1}^{d} \psi_{\gamma_j}(A(f_1) B_j(g_j) B_j(h_{1,j})),$$

we can write

$$|\,\mathrm{E}_1[C_2]|^2 = \left| \sum_{\gamma_1 \in G \setminus \{1_G\}} \cdots \sum_{\gamma_d \in G \setminus \{1_G\}} \mathrm{E}_1[C_3] \right|^2$$

$$\leq \sum_{\gamma_1 \in G \setminus \{1_G\}} \cdots \sum_{\gamma_d \in G \setminus \{1_G\}} |\mathrm{E}_1[C_3]|^2$$

and summarize our calculations so far as

$$|\mathrm{E}[T_S]|^2 \leq (|G|-1)^{|S|-d} \sum_{\gamma_1 \in G \setminus \{1_G\}} \cdots \sum_{\gamma_d \in G \setminus \{1_G\}} \mathrm{E}\Big[|\mathrm{E}_1[C_3]|^2\Big].$$

Thus, there exists some $\gamma_1, \ldots, \gamma_d \in G \setminus \{1_G\}$ such that

$$|\mathrm{E}[T_S]|^2 \leq (|G|-1)^{|S|} \mathrm{E}\Big[|\mathrm{E}_1[C_3]|^2\Big].$$

From now on, we fix these $\gamma_1, \ldots, \gamma_d \in G \setminus \{1_G\}$ and try to bound the corresponding

$$|\mathrm{E}_1[C_3]|^2 = \left|\mathrm{E}_1\left[\prod_{j=1}^{d} \psi_{\gamma_j}(A(f_1)B_j(g_j)B_j(h_{1,j}))\right]\right|^2 \tag{30}$$

by a sum of Fourier coefficients. By the homomorphism property (3) and the fact (4) that ψ_a is linear in a,

$$C_3 = \psi_{\gamma_1 \cdots \gamma_d}(A(f_1)) \prod_{j=1}^{d} \psi_{\gamma_j}(B_j(g_j))\psi_{\gamma_j}(B_j(h_{1,j})).$$

Since $\psi_{\gamma_j}(B_j(g_j))$ are independent of f_1 and $e_{1,1}, \ldots, e_{1,k}$, we can move them outside $E_1[\cdot]$. Since $|\psi_{\gamma_j}(B_j(g_j))| = 1$, this simplifies the expectation (30) to

$$|\mathrm{E}_1[C_3]|^2 = \left|\mathrm{E}_1\left[\psi_{\gamma_1 \cdots \gamma_d}(A(f_1)) \prod_{j=1}^{d} \psi_{\gamma_j}(B_j(h_{1,j}))\right]\right|^2 \tag{31}$$

The remaining factors are expressed using the Fourier transform:

$$\psi_{\gamma_1 \cdots \gamma_d}(A(f_1)) = \sum_{\alpha \in \mathcal{F}_U^G} \hat{A}_\alpha \chi_\alpha(f_1), \tag{32}$$

$$\psi_{\gamma_j}(B_j(h_{1,j})) = \sum_{\beta_j \in \mathcal{F}_{W_j}^G} \hat{B}_{j,\beta_j} \chi_{\beta_j}(h_{1,j}), \tag{33}$$

where

$$\hat{A}_\alpha = \langle \psi_{\gamma_1 \cdots \gamma_d} \circ A, \chi_\alpha \rangle, \qquad \hat{B}_{j,\beta_j} = \langle \psi_{\gamma_j} \circ B_j, \chi_{\beta_j} \rangle.$$

Note that the first of the above inner products is in $L^2(\mathcal{F}_U^G)$ while the latter is in $L^2(\mathcal{F}_{W_j}^G)$. When we insert the Fourier expansions (32) and (33) into the expectation (31) and expand the products, we obtain one term for each possible combination of α and $\beta_1, \dots, \beta_d$:

$$|\mathrm{E}_1[C_3]|^2 = \left| \sum_{\alpha \in \mathcal{F}_U^G} \sum_{\beta_1 \in \mathcal{F}_{W_1}^G} \cdots \sum_{\beta_d \in \mathcal{F}_{W_d}^G} \mathrm{E}_1[\hat{A}_\alpha \hat{B}_{1,\beta_1} \cdots \hat{B}_{d,\beta_d} C_4] \right|^2, \tag{34}$$

where

$$C_4 = \chi_\alpha(f_1)\chi_{\beta_1}(h_{1,1}) \cdots \chi_{\beta_d}(h_{1,d}).$$

Note that the Fourier coefficients can be moved out from $\mathrm{E}_1[\cdots]$ since they are independent of f_1 and $e_{1,1}, \dots, e_{1,k}$. This simplifies the expectation (34) even further to

$$\begin{aligned} |\mathrm{E}_1[C_3]|^2 &= \left| \sum_{\alpha \in \mathcal{F}_U^G} \sum_{\beta_1 \in \mathcal{F}_{W_1}^G} \cdots \sum_{\beta_d \in \mathcal{F}_{W_d}^G} \hat{A}_\alpha \hat{B}_{1,\beta_1} \cdots \hat{B}_{d,\beta_d} \, \mathrm{E}_1[C_4] \right|^2 \\ &\leq \sum_{\alpha \in \mathcal{F}_U^G} \sum_{\beta_1 \in \mathcal{F}_{W_1}^G} \cdots \sum_{\beta_d \in \mathcal{F}_{W_d}^G} |\hat{A}_\alpha|^2 |\hat{B}_{1,\beta_1}|^2 \cdots |\hat{B}_{d,\beta_d}|^2 |\mathrm{E}_1[C_4]|^2 \end{aligned} \tag{35}$$

Fortunately many of the terms in the above sum vanish. Since $h_{ij} = (f_i g_j e_{ij})^{-1}$, it follows from the homomorphism property (11), the fact (12) that χ_α is linear in α, and Lemma 1 that

$$C_4 = \chi_{\alpha(\pi_U(\beta_1)\cdots\pi_U(\beta_d))^{-1}}(f_1) \prod_{j=1}^{d} \chi_{\beta_j}(g_j^{-1}) \chi_{\beta_j}(e_{1,j}^{-1}).$$

Since all factors in the above product are independent, we can take the expectation of each factor separately. From the summation identity (14),

$$\mathrm{E}_1[\chi_{\alpha(\pi_U(\beta_1)\cdots\pi_U(\beta_d))^{-1}}(f_1)] = \begin{cases} 1 & \text{if } \alpha = \pi_U(\beta_1)\cdots\pi_U(\beta_d), \\ 0 & \text{otherwise.} \end{cases}$$

The factors $\chi_{\beta_j}(g_j^{-1})$ are independent of f_1 and $e_{1,1}, \dots, e_{1,k}$, which implies that

$$\mathrm{E}_1[\chi_{\beta_j}(g_j^{-1})] = \chi_{\beta_j}(g_j^{-1}).$$

By the definition of the functions $e_{i,j}$ we obtain

$$\mathrm{E}_1[\chi_{\beta_j}(e_{1,j}^{-1})] = (1 - \delta_1)^{|\beta_j|}.$$

To summarize,

$$|\mathrm{E}_1[C_4]|^2 = \begin{cases} (1 - \delta_1)^{2(|\beta_1| + \cdots + |\beta_d|)} & \text{if } \alpha = \pi_U(\beta_1)\cdots\pi_U(\beta_d), \\ 0 & \text{otherwise.} \end{cases}$$

With this in mind, we can rewrite the expectation (35) as

$$|\mathrm{E}_1[C_3]|^2 \le \sum_{\substack{\alpha,\beta_1,\dots,\beta_d\\ \alpha=\pi_U(\beta_1)\cdots\pi_U(\beta_d)}} |\hat{A}_\alpha|^2|\hat{B}_{1,\beta_1}|^2\cdots|\hat{B}_{d,\beta_d}|^2(1-\delta_1)^{2(|\beta_1|+\cdots+|\beta_d|)}.$$

Thus, there exists some $\gamma_1,\dots,\gamma_d \in G\setminus\{1_G\}$ such that

$$\delta_2^2 \le (|G|-1)^{|S|}\,\mathrm{E}\Big[|\mathrm{E}_1[C_3]|^2\Big] \le (|G|-1)^{|S|} \sum_{\substack{\alpha,\beta_1,\dots,\beta_d\\ \alpha=\pi_U(\beta_1)\cdots\pi_U(\beta_d)}} \mathrm{E}_{U,\Phi_1,\dots,\Phi_d}[Q],$$

where Q is defined as in the formulation of the lemma. ∎

5.6 Designing Efficient Provers

Lemma 10. *Suppose that* $\mathrm{E}[T_S] \ge \delta_2 > 0$ *for some set* $S \ne \emptyset$ *such that* $S \subseteq E$*. Then there exists provers that make the two-prover one-round protocol for* μ*-gap E3-Sat(5) from Sec. 5.1 accept with probability at least* $\delta_1\delta_2^2/(|G|-1)^{|S|}$.

Proof. To construct their strategy, the provers first compute the $\gamma_1,\dots,\gamma_d$ maximizing $\mathrm{E}_{U,\Phi_1,\dots,\Phi_d}[Q]$, where Q is defined as in (23). They then fix these $\gamma_1,\dots,\gamma_d$ for the remaining computation. After these initial preparations, the provers proceed as follows:

Prover P_1 receives a set U of u variables. For $j = 2,\dots,d$, P_1 selects uniformly at random u clauses $C_{j,1},\dots,C_{j,u}$ such that clause $C_{j,i}$ contains variable x_i. Let Φ_j be the Boolean formula $C_{j,1}\wedge\cdots\wedge C_{j,u}$. Let W_j be the set of variables in the clauses $C_{j,1},\dots,C_{j,u}$. Then P_1 computes the Fourier coefficients $\hat{A}_\alpha = \langle\psi_{\gamma_1\cdots\gamma_d}\circ A,\chi_\alpha\rangle$ and $\hat{B}_{j,\beta_j} = \langle\psi_{\gamma_j}\circ B_j,\chi_{\beta_j}\rangle$ for $j = 2,\dots,d$, selects $(\alpha,\beta_2,\dots,\beta_d)$ randomly such that $\Pr[(\alpha,\beta_2,\dots,\beta_d)] = |\hat{A}_\alpha|^2|\hat{B}_{2,\beta_2}|^2\cdots|\hat{B}_{d,\beta_d}|^2$, forms the function $\alpha' = \alpha(\pi_U(\beta_2)\cdots\pi_U(\beta_d))^{-1}$ and returns an arbitrary x such that $\alpha'(x)\ne 1_G$. If no such x exists, P_1 returns an arbitrary $x\in\{-1,1\}^U$.

Prover P_2 receives Φ_1 consisting of u clauses, computes $\hat{B}_{1,\beta_1} = \langle\psi_{\gamma_1}\circ B_j,\chi_{\beta_1}\rangle$, selects a random β_1 with the distribution $\Pr[\beta_1] = |\hat{B}_{1,\beta_1}|^2$, and returns a random y such that $\beta_1(y)\ne 1_G$. By Lemma 2 such a y always exists, and by Lemma 3 such assignments satisfy Φ_1.

Let us now analyze the acceptance probability of this strategy. In the analysis we bound $\Pr[\text{accept} \mid U,\Phi_1,\dots,\Phi_m]$ from below. This is enough to prove the lemma, since

$$\Pr[\text{accept}] = \mathrm{E}[\Pr[\text{accept} \mid U,\Phi_1,\dots,\Phi_d]].$$

Thus, we assume from now on that U and $\Phi_1,\dots,\Phi_d$ are fixed and try to estimate the acceptance probability under these assumptions.

Since Lemma 9 proves a lower bound on $\mathrm{E}[Q]$, we want to express the acceptance probability in terms of Q. Note that Lemma 2 implies that $\alpha \neq 1_G$, since the provers never choose an α such that $\hat{A}_\alpha = 0$, and in the same way Corollary 1 ensures that the selected β_j has the property that $\pi_U(\beta_j) \neq 1_G$. This means, that if the provers obtain $(\alpha, \beta_1, \dots, \beta_d)$ such that $\alpha = \pi_U(\beta_1) \cdots \pi_U(\beta_d)$, there exists x such that $(\pi_U(\beta_1))(x) \neq 1_G$, and for every such x the function $\alpha' = \alpha(\pi_U(\beta_2) \cdots \pi_U(\beta_d))^{-1}$ sends x to an element in $G \setminus \{1_G\}$. Put another way:

$$\alpha'(x) \neq 1_G \iff (\pi_U(\beta_1))(x) \neq 1_G.$$

This implies that there exists a y such that $x = y|_U$ and $\beta_1(y) \neq 1_G$. Given the x chosen by P_1, the probability that P_2 chooses a y such that $y|_U = x$ and $\beta_1(y) \neq 1_G$ is at least $1/|\beta_1|$. All this put together implies that the acceptance probability can be bounded from below by

$$\Pr[\text{accept} \mid U, \Phi_1, \dots, \Phi_m] \geq \sum_{\substack{\alpha, \beta_1, \dots, \beta_d \\ \alpha = \pi_U(\beta_1) \cdots \pi_U(\beta_d)}} \frac{|\hat{A}_\alpha|^2 |\hat{B}_{1,\beta_1}|^2 \cdots |\hat{B}_{d,\beta_d}|^2}{|\beta_1|}.$$

Since $e^x > 1 + x > x$ for any real positive x,

$$\frac{e^{\delta_1 |\beta|}}{\delta_1} > \frac{\delta_1 |\beta|}{\delta_1} = |\beta|,$$

or equivalently,

$$\frac{1}{|\beta|} > \delta_1 e^{-\delta_1 |\beta|} > \delta_1 (1 - \delta_1)^{|\beta|},$$

where the second inequality follows from $e^{-x} > 1 - x$, which is true for any real positive x, we obtain

$$\Pr[\text{accept} \mid U, \Phi_1, \dots, \Phi_m] \geq \delta_1 \sum_{\substack{\alpha, \beta_1, \dots, \beta_d \\ \alpha = \pi_U(\beta_1) \cdots \pi_U(\beta_d)}} |\hat{A}_\alpha|^2 |\hat{B}_{1,\beta_1}|^2 \cdots |\hat{B}_{d,\beta_d}|^2 (1 - \delta_1)^{|\beta_1|}.$$

By Lemma 9, this implies that

$$\Pr[\text{accept} \mid U, \Phi_1, \dots, \Phi_m] \geq \frac{\delta_1 \delta_2^2}{(|G| - 1)^{|S|}}$$

since $0 < \delta_1 < 1$. ∎

5.7 Putting the Pieces Together

Lemma 11. *Suppose that the test in Sec. 5.2 accepts with probability at least $1/|G|^{|E|} + \delta_2$. Then there exist provers that make the two-prover one-round protocol for μ-gap E3-Sat(5) from Sec. 5.1 accept with probability at least $\delta_1\delta_2^2/(|G|-1)^{|E|}$.*

Proof. By Lemma 8 the assumptions in the lemma implies that $|\operatorname{E}[T_S]| \geq \delta_2$ for some $S \neq \emptyset$ such that $S \subseteq E$. By Lemmas 9 and 10, this implies that there exists provers that make the two-prover one-round protocol for μ-gap E3-Sat(5) from Sec. 5.1 accept with probability at least $\delta_1\delta_2^2/(|G|-1)^{|S|}$. Since $|S| \leq |E|$, the lemma follows. ∎

Lemma 12. *For every constant $\delta_2 > 0$, it is possible to select a constant u such that the soundness of the PCP in Sec. 5.2 is at most $1/|G|^{|E|} + \delta_2$.*

Proof. Suppose that the PCP in Sec. 5.2 has soundness $1/|G|^{|E|}+\delta_2$ for some constant $\delta_2 > 0$. By Lemma 11, this implies that the two-prover one-round interactive proof system for μ-gap E3-Sat(5) has soundness $\delta_1\delta_2^2/(|G|-1)^{|E|}$. But we know [7] that the soundness of this proof system is at most c_μ^u, where $c_\mu < 1$ is a constant and u is the cardinality of U. If we select

$$u > \frac{\log \delta_1^{-1}\delta_2^{-2} + \log(|G|-1)^{|E|}}{\log c_\mu^{-1}},$$

note that this latter quantity is a constant since δ_1, δ_2, $|E|$, $|G|$, and c_μ are constants, we obtain

$$\frac{\delta_1\delta_2^2}{(|G|-1)^{|E|}} > c_\mu^u,$$

which is a contradiction. ∎

6 The Reduction to Non-Boolean CSPs

We now show how the above PCP can be connected with CSPs to prove that the corresponding CSPs are non-approximable beyond the random assignment threshold. As for the completeness c and the soundness s of the PCP from the previous section, we have shown that $c \geq (1-\delta_1)^{|E|}$ and $s \leq |G|^{-|E|} + \delta_2$ for arbitrarily small constants $\delta_1, \delta_2 > 0$.

Theorem 2. *Let G be any finite Abelian group, ℓ and m be arbitrary positive integers, and $E \subseteq [\ell] \times [m]$. Then the predicate*

$$\bigwedge_{i,j:(i,j)\in E} (x_i x_j x_{i,j} = a_{i,j}),$$

where $a_{i,j} \in G$ and x_i, x_j, and $x_{i,j}$ assume values in G, is non-approximable beyond the random assignment threshold.

Before proving the theorem, we restate it in slightly different words.

Definition 15. Max k-CSP-G *is the following maximization problem: Given a number of functions from G^k, where G is a finite Abelian group, to $\boldsymbol{Z}_2$, find the assignment maximizing the number of functions evaluating to* 1. *The total number of variables in the instance is denoted by n.*

Theorem 2, rephrased. *Let G be any finite Abelian group, ℓ and m be arbitrary positive integers, $E \subseteq [\ell] \times [m]$, and $k = |E| + \ell + m$. Then it is **NP**-hard to approximate Max k-CSP-G within $|G|^{|E|} - \epsilon$ for any constant $\epsilon > 0$.*

Proof. Select the constants $\delta_1 > 0$ and $\delta_2 > 0$ such that

$$\frac{(1-\delta_1)^{|E|}}{|G|^{-|E|} + \delta_2} \geq |G|^{|E|} - \epsilon.$$

Then select the constant u such that $\delta_1 \delta_2^2 / (|G| - 1)^{|E|} > c_\mu^u$. Now consider applying the PCP from Sec. 5.2 to an instance of the **NP**-hard problem μ-gap E3-Sat(5).

Construct an instance of Max k-CSP-G as follows: Introduce variables $x_{U,f}$ and $y_{\Phi_j,g}$ for every $A(f)$ and $B_j(g)$, respectively. For all possible combinations of a set U, clauses $\Phi_1, \dots, \Phi_m$, and functions $f_1, \dots, f_\ell$, $g_1, \dots, g_m$, $h_{1,1}, \dots, h_{\ell,m}$, introduce a constraint that is one if $x_{U,f_i} y_{\Phi_j,g_j} = y_{\Phi_j,h_{ij}}$ for all $(i,j) \in E$. Set the weight of this constraint to the probability of the event that the set U, the clauses $\Phi_1, \dots, \Phi_m$, and the functions $f_1, \dots, f_\ell$, $g_1, \dots, g_m$, and $h_{1,1}, \dots, h_{\ell,m}$ are chosen by the verifier in the PCP. Each constraint is a function of at most $|E| + \ell + m$ variables. The total number of constraints is at most

$$n^u 5^{mu} |G|^{\ell 2^u + m 2^{3u} + \ell m 2^{3u}},$$

which is polynomial in n if ℓ, m, $|G|$, and u are constants. The weight of the satisfied equations for a given assignment to the variables is equal to the probability that the PCP from Sec. 5.2 accepts the proof corresponding to this assignment. Thus, any algorithm approximating the optimum of the above instance within

$$\frac{(1-\delta_1)^{|E|}}{|G|^{-|E|}+\delta_2} \geq |G|^{|E|} - \epsilon$$

decides the **NP**-hard problem μ-gap E3-Sat(5). ∎

Corollary 2. *For any integer $k \geq 3$ and any constant $\epsilon > 0$, it is **NP**-hard to approximate Max k-CSP-G within $|G|^{k-2\sqrt{k+1}+1} - \epsilon$.*

Proof. As a warmup, assume that $k = s^2 + 2s$ for some positive integer s. Then we can choose $\ell = m = s$ and $E = [\ell] \times [m]$ in Theorem 2 and obtain that it is **NP**-hard to approximate Max k-CSP-G within $|G|^{s^2} - \epsilon$, for any constant $\epsilon > 0$. To express this as a function of k, note that

$$k = s^2 + 2s \iff s = \sqrt{k+1} - 1,$$

which implies that

$$s^2 = k + 1 + 1 - 2\sqrt{k+1} = k - 2\sqrt{k+1} + 2.$$

Thus, it is **NP**-hard to approximate Max k-CSP-G within $|G|^{k-2\sqrt{k+1}+2} - \epsilon$, for any constant $\epsilon > 0$, when $k = s^2 + 2s$ for some positive integer s. To investigate what happens when $s^2 + 2s < k < (s+1)^2 + 2(s+1) = s^2 + 4s + 3$ we proceed in two stages.

In the first stage, we assume that $k = s^2 + 2s + 1$ where s is an arbitrary positive integer. In that case, we can set $\ell = s$, $m = s + 1$, and E to any subset of $[\ell] \times [m]$ containing s^2 edges. Then Theorem 2 implies that it is **NP**-hard to approximate Max k-CSP-G within $|G|^{s^2} - \epsilon$, for any constant $\epsilon > 0$, in this special case. Then we rewrite this as a function of k by using the relation

$$k = s^2 + 2s + 1 \iff s = \sqrt{k} - 1,$$

which implies that

$$s^2 = k - 2\sqrt{k} + 1.$$

Thus, it is **NP**-hard to approximate Max k-CSP-G within $|G|^{k-2\sqrt{k}+1} - \epsilon$, for any constant $\epsilon > 0$, when $k = s^2 + 2s + 1$ for some positive integer s.

In the second stage, we assume that $k = s^2 + 2s + 2 + t$ where s is an arbitrary positive integer and t is an integer satisfying $0 \leq t \leq 2s$. In that case, we can set $\ell = m = s + 1$ and let E be any subset of $[\ell] \times [m]$ containing $s^2 + t$ edges. Then Theorem 2 implies that it is **NP**-hard to approximate

Max k-CSP-G within $|G|^{s^2+t} - \epsilon$, for any constant $\epsilon > 0$, in this special case. To express this as a function of k, note that

$$k = s^2 + 2s + 2 + t \iff s = \sqrt{k - t + 1} - 1,$$

which implies that

$$s^2 + t = k - 2\sqrt{k - t + 1} + 2 \geq k - 2\sqrt{k + 1} + 2.$$

Thus, it is **NP**-hard to approximate Max k-CSP-G within $|G|^{k-2\sqrt{k+1}+2} - \epsilon$, for any constant $\epsilon > 0$, when $s^2 + 2s + 2 \leq k \leq s^2 + 4s + 2$ for some positive integer s. Therefore, it is **NP**-hard to approximate Max k-CSP-G within $|G|^{k-2\sqrt{k+1}+1} - \epsilon$, for any constant $\epsilon > 0$ and any positive integer $k \geq 3$.

∎

From the details of the proof of Corollary 2, we see that we can rephrase the result in a slightly stronger form.

Corollary 3. *For any integer $s \geq 2$ and any constant $\epsilon > 0$, it is **NP**-hard to approximate Max s^2-CSP-G within $|G|^{(s-1)^2} - \epsilon$. For any integer $k \geq 3$ that is not a square and any constant $\epsilon > 0$, it is **NP**-hard to approximate Max k-CSP-G within $|G|^{k-2\sqrt{k+1}+2} - \epsilon$.*

7 Conclusions

We have shown that it is possible to combine the harmonic analysis introduced by Håstad [6] with the recycling techniques used by Samorodnitsky and Trevisan [8] to obtain a lower bound on the approximability of Max k-CSP-G. The proof of results of this type typically study some predicate on a constant number of variables such that a random assignment to the variables satisfies the predicate with probability $1/w$. Starting from the 2P1R interactive proof system for μ-gap E3-Sat(5) reviewed in Sec. 5.1, instances such that it is **NP**-hard to approximate the number of satisfied constraints within $w - \epsilon$, for any constant $\epsilon > 0$, are constructed. Our proof is no exception to this rule.

The current state of the art regarding the (non-)approximability of predicates is that there are a number of predicates—such as linear equations mod p with three unknowns in every equation, E3-satisfiability, and the predicates of this paper—that are non-approximable beyond the random assignment threshold [6, 8]. There also exists some predicates—such as linear equations mod p with two unknowns in every equation and E2-satisfiability—where there are polynomial time algorithms beating the bound obtained from a random assignment [1, 4, 5].

A very interesting direction for future research would be to try to determine criteria identifying predicates that are non-approximable beyond the random assignment threshold. Some such attempts have been made for special cases. For predicates of three Boolean variables, it is known that the predicates that are non-approximable beyond the random assignment threshold are precisely those that are implied by parity [6, 14]. However, the general question remains completely open.

Acknowledgments

The author thanks Johan Håstad for many clarifying discussions on the subject of this paper.

References

1. Gunnar Andersson, Lars Engebretsen, and Johan Håstad. A new way of using semidefinite programming with applications to linear equations mod p. *Journal of Algorithms*, 39(2):162–204, May 2001.

2. Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Márió Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM*, 45(3):501–555, May 1998.

3. Uriel Feige. A threshold of $\ln n$ for approximating set cover. *Journal of the ACM*, 45(4):634–652, July 1998.

4. Uriel Feige and Michel X. Goemans. Approximating the value of two prover proof systems, with applications to MAX 2SAT and MAX DICUT. In *Proceedings of 3rd Israel Symposium on Theory of Computing and Systems*, pages 182–189. 1995.

5. Michel X. Goemans and David P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42(6):1115–1145, November 1995.

6. Johan Håstad. Some optimal inapproximability results. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, pages 1–10. El Paso, Texas, 4–6 May 1997. Accepted for publication in *Journal of the ACM*.

7. Ran Raz. A parallel repetition theorem. *SIAM Journal on Computing*, 27(3):763–803, June 1998.

8. Alex Samorodnitsky and Luca Trevisan. A PCP characterization of NP with optimal amortized query complexity. In *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*, pages 191–199. Portland, Oregon, 21–23 May 2000.

9. Maria Serna, Luca Trevisan, and Fatos Xhafa. The (parallel) approximability of non-Boolean satisfiability problems and restricted integer programming. In Michel Morvan, Christoph Meinel, and Daniel Krob, editors, *Proceedings of the 15th Annual Symposium on Theoretical Aspects of Computer Science*, volume 1373 of *Lecture Notes in Computer Science*, pages 488–498. Springer-Verlag, Paris, 25–27 February 1998.

10. Madhu Sudan and Luca Trevisan. Probabilistically checkable proofs with low amortized query complexity. In *39th Annual Symposium on Foundations of Computer Science*, pages 18–27. IEEE, Palo Alto, California, 8–11 November 1998.

11. Audrey Terras. *Fourier Analysis on Finite Groups and Applications*, volume 43 of *London Mathematical Society student texts*. Cambridge University Press, Cambridge, 1999.

12. Luca Trevisan. Parallel approximation algorithms by positive linear programming. *Algorithmica*, 21(1):72–88, May 1998.

13. Luca Trevisan. Recycling queries in PCPs and in linearity tests. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, pages 299–308. Dallas, Texas, 23–26 May 1998.

14. Uri Zwick. Approximation algorithms for constraint satisfaction programs involving at most three variables per constraint. In *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 201–210. San Francisco, California, 25–27 January 1998.

ECCC ISSN 1433-8092
http://www.eccc.uni-trier.de/eccc
ftp://ftp.eccc.uni-trier.de/pub/eccc
ftpmail@ftp.eccc.uni-trier.de, subject 'help eccc'