THEORY OF COMPUTING, Volume 0 (2005), pp. 1–30
http://theoryofcomputing.org

Query efficient PCPs with perfect completeness

Johan Håstad[*] Subhash Khot[†]

Received: October 21, 2004; published: September ??, 2005.

Abstract: For every integer $k > 0$, and an arbitrarily small constant $\varepsilon > 0$, we present a PCP characterization of NP where the verifier uses logarithmic randomness, non-adaptively queries $4k + k^2$ bits in the proof, accepts a correct proof with probability 1, i.e., it is has perfect completeness, and accepts any supposed proof of a false statement with probability at most $2^{-k^2} + \varepsilon$. In particular, the verifier achieves optimal *amortized query complexity* of $1 + \delta$ for arbitrarily small constant $\delta > 0$. Such a characterization was already proved by Samorodnitsky and Trevisan [12], but their verifier loses perfect completeness and their proof makes an essential use of this feature.

By using an adaptive verifier, we can decrease the number of query bits to $2k + k^2$, the same number obtained in [12]. Finally we extend some of the results to PCPs over non-Boolean alphabets.

[*] Royal Institute of Technology, Stockholm. Work done while visiting the Institute for Advanced Study, supported by NSF grant CCR-9987077.

[†] Georgia Institute of Technology, Atlanta. Work done while the author was at Princeton University.

ACM Classification: F 2.2

AMS Classification: 68Q25

Key words and phrases: PCP, inapproximability, amortized query bits, perfect completeness

1 Introduction

The celebrated PCP Theorem ([2], [1]) gives a way of writing proofs for (purported) NP statements such that the proofs can be checked very efficiently by a probabilistic verifier. The verifier needs a very limited amount of random bits and reads only a constant number of bits from the proof. Moreover, a correct statement always has a proof that is accepted with probability 1 (or close to 1) and *any* proof of an incorrect statement is accepted only with a tiny probability (called error probability or soundness).

PCPs have surprising connections, first discovered by Feige et al. [5], to inapproximability results, i.e., results showing that computing even approximate solutions to some NP-complete problems is hard. The discovery of the PCP Theorem opened up a whole new fascinating direction for proving various inapproximability results. In the last decade or so, quantitative improvement in the efficiency of PCP verifiers has led to (in many cases optimal) inapproximability results for many optimization problems ([3], [4], [14], [13], [12], [6]). For different applications, different aspects of the given PCP need to be optimized. For a detailed discussion of various parameters we refer to [3].

In the current paper we are mostly concerned with making efficient use of queries, i.e., to obtain very strong PCPs where the verifier reads very few symbols in the proof. More specifically, we are interested in the trade-off between the number of queries and the error probability.

Samorodnitsky and Trevisan [12] obtained very strong results along these lines, giving a PCP where the verifier reads $2k+k^2$ bits, almost always accepts a correct proof of a correct statement and accepts a proof of an incorrect statement with probability only marginally larger than 2^{-k^2}. This is a very impressive result in that each read bit essentially decreases the probability of being fooled by a factor of 2. Their verifier achieves *amortized query complexity* of $1+\delta$ for any $\delta > 0$ which is optimal (see [3]). The amortized query complexity, when we (almost) always accept a correct proof, is formally defined as the ratio between the number of queries ($2k+k^2$ in this case) and the logarithm of inverse of the error probability (k^2 in this case).

The fact that the verifier sometimes rejects a correct proof of a correct statement is called imperfect completeness and in their construction Samorodnitsky and Trevisan make essential use of this property of the verifier. For many reasons it is preferable to have perfect completeness. Firstly, it is natural to have a proof system where a correct proof of a correct statement is always accepted. Secondly, perfect completeness is sometimes essential to obtain further results. Some inapproximability results such as graph coloring sometimes make essential use of perfect completeness and when using a given protocol as a subprotocol in future protocols, perfect completeness, to say the least, simplifies matters.

Several results in the past have focused on achieving PCPs with perfect completeness and this task many times turns out to be harder than obtaining corresponding PCPs without this property. For instance, Håstad shows that 3SAT and 4-Set Splitting are hard to approximate within ratio $\frac{8}{7}+\varepsilon$. These results follow from the basic 3-bit PCP of [13] establishing hardness for approximating the number of satisfied linear equations mod 2. To extend these results to satisfiable instances however requires a new PCP construction and a technically more complicated proof.

The main result of the current paper is to extend the result of Samorodnitsky and Trevisan to include perfect completeness.

Theorem 1.1. *For any integer $k > 0$ and any $\varepsilon > 0$, any language in NP has a PCP verifier that queries $4k+k^2$ bits, has perfect completeness and accepts a proof of an incorrect statement with probability at* *most* $2^{-k^2} + \varepsilon$.

Our result is based on a basic non-linear test which reads 5 bits $(b_1, b_2, b_3, b_4, b_5)$ from the proof and accepts if $b_1 = b_2 \oplus b_3 \oplus (b_4 \wedge b_5)$. We call this constraint Tri-Sum-And and let MAX-TSA be the problem of satisfying maximum number of such constraints. We have the following theorem.

Theorem 1.2. *For any $\varepsilon > 0$, it is NP-hard to distinguish satisfiable instances of Max-TSA from those where it is only possible to simultaneously satisfy a fraction $\frac{1}{2} + \varepsilon$ of the constraints.*

The choice to study Tri-Sum-And is somewhat arbitrary but guided by our goal to achieve perfect completeness while keeping the analysis simple. To get perfect completeness we need a nonlinear predicate while the analysis is greatly aided by having as much linearity as possible present in the predicate. These two conflicting requirements led to the choice of Tri-Sum-And.

Note that Theorem 1.2 is tight for Max-TSA in that a random assignment satisfies half the constraints. There are stronger results for other constraints on 5 bits and in particular Guruswami et al. [7] give a different predicate for which $\frac{1}{2}$ can be improved to $\frac{7}{16}$.

We then iterate the basic test underlying Theorem 1.1 in a way similar to Samorodnitsky and Trevisan iterate the basic 3-bit test by Håstad. We present two iterated tests : One which we call the "complete bipartite graph PCP," is analyzed in a way analogous to Samorodnitsky-Trevisan and the other, which we call the "almost disjoint sets PCP," is analyzed in a way analogous to how Håstad and Wigderson [15] analyzed the test of Samorodnitsky and Trevisan.

By a standard reduction the PCP results imply the following theorem.

Theorem 1.3. *Boolean constraint satisfaction problem on k variables is hard to approximate within ratio $2^{k-O(\sqrt{k})}$ on satisfiable instances.*

This should be contrasted with the approximation algorithm by Trevisan [16] that shows that it is possible to approximate Boolean constraint satisfaction problem on k variables within $O(2^k/k)$ on satisfiable instances.

A test is called non-adaptive if which bits to read are decided before the first bit is read and hence this set is independent of the actual proof. All the above mentioned PCPs are non-adaptive which is in fact necessary to obtain Theorem 1.3.

If we allow adaptive tests then by making an iterated version of a test in [7] we can get essentially the same parameters as Samorodnitsky and Trevisan and thus simply gain perfect completeness.

Theorem 1.4. *For any integer $k > 0$ and any $\varepsilon > 0$, any language in NP has an adaptive PCP verifier that queries $2k + k^2$ bits, has perfect completeness and accepts a proof of an incorrect statement with probability at most $2^{-k^2} + \varepsilon$.*

If we convert the test to be non-adaptive, this test would read $2k + 2k^2$ different bits and hence this result does not strictly dominate Theorem 1.1.

We extend some of our results to non-Boolean domains and in particular we have the following theorem.

Theorem 1.5. *For every prime p, the constraint satisfaction problem on k variables over an alphabet of size p is hard to approximate within ratio $p^{k-O(\sqrt{k})}$ on satisfiable instances.*

We hope that our results will be useful in future to prove strong hardness results for approximate graph coloring. One such result by Khot [9] is

Theorem 1.6. *[9] There is an absolute constant $c > 0$ such that it is NP-hard to color a k-colorable graph with $k^{c \log k}$ colors.*

Actually this result can be proved from the original form of the Samorodnitsky-Trevisan's result and perfect completeness is not strictly required. But using our PCP with perfect completeness, this result becomes more straightforward. On a related note one can observe that perfect completeness is essential in the hypergraph coloring results by Guruswami, Håstad and Sudan [6], and in general it is a subtle problem which coloring inapproximability results require perfect completeness in the underlying PCP.

1.1 Overview of the paper

This is the complete version of the extended abstract [8]. The paper is organized as follows. Section 2 introduces techniques used in this paper. In Section 3 we give our results for the Boolean case: Section 3.1 gives our basic 5-bit test, and Section 3.2 describes our iterated tests. Section 4 extends some of the results of Section 3 to non-Boolean domains. Section 5 concludes with a few remarks.

2 The general setup

In this section we provide the necessary background.

2.1 Notation

Throughout the paper, we have Boolean functions in ± 1 notation with -1 as logical true. We use multiplication to denote exclusive-or, $\wedge$ for the logical AND function. As we use -1 to denote true we have

$$x \wedge y = \mathrm{AND}(x,y) = \frac{1+x+y-xy}{2}.$$

Our default is that AND is highest level connective and in particular

$$xy \wedge zw = (xy) \wedge (zw).$$

Addition is used only over the real and complex numbers.

2.2 The 2-prover protocol

Many efficient PCPs, such as the one given in [12] are conveniently analyzed using the formalism of an outer and inner verifier. This could also be done here, but to avoid too much formalism we give a more explicit analysis. Using the results of [1] (as explicitly done in [4]) one can prove that there is a constant $c < 1$ such that it is NP-hard to distinguish satisfiable 3-SAT formulas from those where only a fraction c of the clauses can be simultaneously satisfied by any assignment. This formula can furthermore have the property that any clause is of length exactly 3 and any variable appears in exactly 5 clauses.

Given a 3-SAT formula $\varphi = C_1 \wedge C_2 \ldots \wedge C_m$ which is either satisfiable or where one can only satisfy a fraction c of the clauses, one can design a two-prover interactive proof system with verifier V as follows.

Basic two-prover protocol

1. V chooses a clause C_k uniformly at random and a variable x_j, again uniformly at random, appearing in C_k. V sends k to prover P_1 and j to prover P_2.

2. V receives a value for x_j from P_2 and values for all variables appearing in C_k from P_1. V accepts if the two values for x_j agree and the clause C_k is satisfied.

It is not difficult to see that if a fraction c of the clauses can be satisfied simultaneously then the optimal strategy of P_1 and P_2 convinces V with probability $(2+c)/3$. Thus it is NP-hard to distinguish the case when this probability is 1 and when it is some constant strictly smaller than 1. Note also that if we start with a formula where each variable appears the same number of times, V could first choose a random variable and then a random clause containing that variable and get the same distribution.

To make the gap larger, one runs this protocol u times in parallel resulting in the following protocol.

u-parallel two-prover protocol, 2PP(u)

1. V chooses u clauses $(C_{k_i})_{i=1}^u$ uniformly at random and for each i, V chooses a variable x_{j_i}, again uniformly at random, appearing in C_{k_i}. V sends $(k_i)_{i=1}^u$ to prover P_1 and $(j_i)_{i=1}^u$ to prover P_2.

2. V receives values for $(x_{j_i})_{i=1}^u$ from P_2 and values for all variables appearing in $(C_{k_i})_{i=1}^u$ from P_1. V accepts if the two values for x_{j_i} agree for each i and all the picked clauses are satisfied.

We let U denote the set of variables sent to P_2, i.e., $(x_{j_i})_{i=1}^u$ while the set of variables that P_1 gives values to is denoted by W. Note that $U \subset W$.

By the fundamental result by Raz [11], the probability that the verifier accepts in 2PP(u) when only a constant fraction $c < 1$ of the clauses can be simultaneously satisfied is bounded by d_c^u for some absolute constant $d_c < 1$. Let us formulate these properties for future reference.

Theorem 2.1. *Let 2PP(u) be the u parallel version of the basic two-prover protocol. Then if only a fraction $c < 1$ of the clauses of φ can be simultaneously satisfied, then no strategy of P_1 and P_2 can make the verifier accept with probability greater than d_c^u. Here $d_c < 1$ is a constant that only depends on c.*

2.3 Long codes

To turn the protocol 2PP(u) into a written proof that can be checked very efficiently, it is natural to, for each question to either P_1 or P_2, write down the answer in coded form. As many other papers we use the *l*ong code introduced by Bellare et al [3].

Definition 2.2. The *l*ong code of an assignment $x \in \{-1,1\}^t$ is obtained by writing down for each function $f : \{-1,1\}^t \to \{-1,1\}$, the value $f(x)$.

Thus the long code of a string of length t is a string of length 2^{2^t}. Note that even though a prover is supposed to write down a long code for an assignment a cheating prover might write down a string which is not the correct long code of anything. We analyze such arbitrary tables by the Fourier-expansion.

2.3.1 Fourier Analysis

In this section, we explain the basics of the Fourier method. Let

$$\mathcal{F} = \{f \mid \{-1,1\}^t \to \{-1,1\}\}$$

and consider the vector space of all "tables" $A : \mathcal{F} \to \mathbb{R}$. Here the addition of two tables is defined as pointwise addition and the dimension of this vector space is $|\mathcal{F}| = 2^{2^t}$. One can define a natural inner product on this space by letting the inner product of two tables A_1 and A_2 be

$$< A_1, A_2 > = \; 2^{-2^t} \sum_f A_1(f) A_2(f)$$

For $\alpha \subseteq \{-1,1\}^t$, let χ_α be a Boolean-valued (i.e., $\{-1,1\}$-valued) table defined as

$$\chi_\alpha(f) = \prod_{x \in \alpha} f(x) \quad \forall\, f \,\in \mathcal{F}$$

The χ_α are called *c*haracters. The characters are multiplicative, i.e.,

$$\chi_\alpha(f_1 f_2) = \chi_\alpha(f_1)\chi_\alpha(f_2).$$

The characters are in fact symmetric in α and f but as we have used set notation for α we have

$$\chi_{\alpha_1}(f)\chi_{\alpha_2}(f) = \chi_{\alpha_1 \oplus \alpha_2}(f) \tag{2.1}$$

where $\alpha_1 \oplus \alpha_2$ is the exclusive-or of the characteristic vectors of the sets α_1 and α_2. Put differently, $\alpha_1 \oplus \alpha_2$ is the set which is the symmetric difference of α_1 and α_2.

The set of characters (there are 2^{2^t} of them) forms an orthonormal basis for the vector space. Thus any table A can be expressed as

$$A(f) \;=\; \sum_{\alpha \subseteq \{-1,1\}^t} \hat{A}_\alpha \chi_\alpha(f),$$

where $\hat{A}_\alpha$ are real numbers called *F*ourier coefficients; they can found as

$$\hat{A}_\alpha \;= < A, \chi_\alpha >= \; 2^{-2^t} \sum_f \hat{A}(f) \chi_\alpha(f).$$

If A is Boolean valued, we have Parseval's identity $\sum_\alpha \hat{A}_\alpha^2 = 1$. If A is indeed a correct long code of a string $x^{(0)}$ then $\hat{A}_{\{x^{(0)}\}} = 1$ while all the other Fourier coefficients are 0.

In our protocols we pick function uniformly and then often perform an analysis using the Fourier expansion. The following lemma is simple but powerful.

Lemma 2.3. *Assume that f is picked with the uniform distribution then for $\alpha \neq \emptyset$,*

$$E_f[\chi_\alpha(f)] = 0$$

while

$$E_f[\chi_\emptyset(f)] = 1.$$

Using this lemma together with (2.1) enables us to compute the expected value of products of characters in a simple way.

We can, to a limited extent, put some restrictions on the tables produced by the prover.

Definition 2.4. A table A is folded over true if $A(f) = -A(-f)$ for any f.

Definition 2.5. A table A is conditioned upon a function $h : \{-1,1\}^t \to \{-1,1\}$, if $A(f) = A(f \wedge h)$ for any f.

To make sure that an arbitrary table is folded we access the table as follows. For each pair $(f,-f)$ we choose (in some arbitrary but fixed way) one representative. If f is chosen, then if the value of the table is required at f it is accessed the normal way by reading $A(f)$. If the value at $-f$ is required then in this case also $A(f)$ is read but the result is negated. If $-f$ is chosen from the pair the procedures are reversed.

Similarly we can make sure that a given table is properly conditioned by always reading $A(f \wedge h)$ when the value for f is needed. Folding over true and conditioning can be done at the same time.

Let us now give the consequences of folding and conditioning for the Fourier coefficients. The proofs are easy and left to the reader but they can also be found in [14].

Lemma 2.6. *If A is folded over true and $\hat{A}_\alpha \neq 0$ then $|\alpha|$ is odd and in particular α is non-empty.*

Lemma 2.7. *If A is conditioned upon h and $\hat{A}_\alpha \neq 0$ then for every $x \in \alpha$, $h(x)$ is true (i.e., $h(x) = -1$).*

We will be working with sets U and W with the property that $U \subset W$ and we let $\pi : \{-1,1\}^W \to \{-1,1\}^U$ be the projection operator that maps an assignment on W to its subassignment on U. For every $\beta \subseteq \{-1,1\}^W$, let $\pi(\beta) \subseteq \{-1,1\}^U$ be defined as

$$\pi(\beta) = \{\pi(y) \mid y \in \beta\}.$$

We also need an operator π_2 defined as follows : for any $\beta \subseteq \{-1,1\}^W$, $\pi_2(\beta) \subseteq \{-1,1\}^U$ is the set of those x which have an odd number of preimages in β, i.e.,

$$\pi_2(\beta) = \{x \mid x \in \{-1,1\}^U,\ |\beta \cap \pi^{-1}(x)| \text{ is odd}\}.$$

Note that these projection operators depend on the identities of U and W but as no confusion is likely to arise we suppress this fact.

A function f with domain $\{-1,1\}^U$ can naturally be extended to domain $\{-1,1\}^W$ by simply using the value $f(\pi(y))$. We use the same symbol to denote this extended function and hope that no confusion arises. We have the following simple lemma.

Lemma 2.8. *Let $\beta \subseteq \{-1,1\}^W$, $U \subseteq W$ and $f : \{-1,1\}^U \to \{-1,1\}$, then*

$$\chi_\beta(f) = \chi_{\pi_2(\beta)}(f).$$

3 Efficient PCPs for Boolean domains

In this section we convert 2PP(u) to a PCP. We eliminate the provers by asking the prover to write down the answer to each question (in encoded form). Furthermore, remember that U is the set of u variables which are sent to P_2 in the two-prover protocol. For each possible set U we ask the prover to write a table, A_U, which is supposed to be the long code of the answer by P_2 on question U. We assume that A_U is folded over true.

Similarly W is the set of variables in the u clauses sent to P_1 and let φ_W be the conjunction of the clauses chosen. In the PCP we have a table, B_W, which is a supposed to be the long code of the answer of P_1 on question W. We assume that B is folded over true and conditioned upon φ_W.

3.1 Our basic test

We have the following basic test, defined using the conventions above.

Basic PCP

1. V chooses U, W and φ_W as in 2PP(u).
2. V chooses two functions f and f' on U uniformly at random (i.e., $f, f' : \{-1,1\}^U \to \{-1,1\}$).
3. V chooses two functions g and g' on W uniformly at random (i.e., $g, g' : \{-1,1\}^W \to \{-1,1\}$). V defines a third function h by setting, for each $y \in \{-1,1\}^W$, $h(y) = g(y)f(\pi(y))(g'(y) \wedge f'(\pi(y)))$.
4. V accepts iff $B_W(h) = B_W(g)A_U(f)(B_W(g') \wedge A_U(f'))$.

We have the basic completeness lemma.

Lemma 3.1. *The completeness of the basic PCP is 1.*

Proof. In a correct proof of a correct theorem each table is a correct long code of a restriction of a given global assignment to the set in question. If we denote this assignment by z then $B_W(h) = h(\pi^W(z))$ where π^W is the projection onto W and similarly for the other involved functions. The completeness now follows from the definition of h. □

The main problem is to establish soundness.

Lemma 3.2. *If the verifier in the basic test accepts with probability $(1+\delta)/2$ then there exists a strategy for P_1 and P_2 in 2PP(u) that makes the verifier accept with probability $\delta^{O(1)}$. In particular if the protocol 2PP(u) is chosen to have sufficiently small soundness (by choosing u large enough), then the verifier in the basic test accepts with probability at most $(1+\delta)/2$.*

Proof. For readability we drop the subscripts and use A instead of A_U and B instead of B_W. Consider the expression

$$\frac{1+B(h)B(g)A(f)(B(g') \wedge A(f'))}{2}.$$

This expression is 1 if the test accepts and 0 otherwise. Hence the probability of acceptance for the test is the expectation of this expression over the choice of f, f', g, g', U, and W. The hypothesis of the lemma implies that

$$E_{f,f',g,g',U,W}[B(h)B(g)A(f)(B(g')\wedge A(f'))] = \delta. \tag{3.1}$$

Fix U, W, f' and g' and let us study

$$E_{f,g}[B(h)B(g)A(f)].$$

Replacing each function by its Fourier expansion we see that this equals

$$\sum_{\beta_1,\beta_2,\alpha} \hat{B}_{\beta_1}\hat{B}_{\beta_2}\hat{A}_\alpha E_{f,g}[\chi_{\beta_1}(fg(f'\wedge g'))\chi_{\beta_2}(g)\chi_\alpha(f)].$$

which, using (2.1) and Lemma 2.8, can be simplified to

$$\sum_{\beta_1,\beta_2,\alpha} \hat{B}_{\beta_1}\hat{B}_{\beta_2}\hat{A}_\alpha E_{f,g}[\chi_{\beta_1}(f'\wedge g')\chi_{\beta_1\oplus\beta_2}(g)\chi_{\pi_2(\beta_1)\oplus\alpha}(f)]. \tag{3.2}$$

Using Lemma 2.3, the inner expected value is 0 unless $\beta_1 = \beta_2 = \beta$ and $\pi_2(\beta) = \alpha$ and otherwise it is 1. Thus the expected value in (3.2) equals

$$\sum_\beta \hat{B}_\beta^2 \hat{A}_{\pi_2(\beta)}\chi_\beta(f'\wedge g'),$$

and hence we need to analyze

$$E_{f',g'}[\chi_\beta(f'\wedge g')(B(g')\wedge A(f'))]. \tag{3.3}$$

We have $a\wedge b = \frac{1}{2}(1+a+b-ab)$ and thus (3.3) equals

$$\frac{1}{2}\left(E[\chi_\beta(f'\wedge g')]+E[\chi_\beta(f'\wedge g')B(g')]+E[\chi_\beta(f'\wedge g')A(f')]-E[\chi_\beta(f'\wedge g')B(g')A(f')]\right) \tag{3.4}$$

Fix the value of f' and let

$$\beta' = \{y \mid y\in\beta \ \wedge \ f'(\pi(y)) = -1\}.$$

When averaging over g', the first and third expected values in (3.4) are 0 unless $\beta' = \emptyset$ while the second and the fourth expected values equal $\hat{B}_{\beta'}$ and $\hat{B}_{\beta'}A(f')$, respectively. To estimate the first and third terms we note that the probability, over the choice of f', that β' is empty is $2^{-|\pi(\beta)|}$. For the other terms we set

$$\alpha = \{x \mid x\in\pi(\beta) \ \wedge \ f'(x) = -1\}$$

and use the Cauchy-Schwartz inequality to obtain

$$E_{f'}\left[|\hat{B}_{\beta'}|\right] = 2^{-|\pi(\beta)|}\sum_{\alpha\subseteq\pi(\beta)}|\hat{B}_{\beta\cap\pi^{-1}(\alpha)}| \le 2^{-|\pi(\beta)|/2}\left(\sum_{\alpha\subseteq\pi(\beta)}\hat{B}^2_{\beta\cap\pi^{-1}(\alpha)}\right)^{1/2} \le 2^{-|\pi(\beta)|/2}. \tag{3.5}$$

This implies that we get an overall upper bound on the left hand side of (3.1) as

$$E_{U,W}\left[\sum_{\beta}\hat{B}_{\beta}^{2}|\hat{A}_{\pi_2(\beta)}|(2^{-|\pi(\beta)|}+2^{-|\pi(\beta)|/2})\right] \le E_{U,W}\left[\sum_{\beta}\hat{B}_{\beta}^{2}|\hat{A}_{\pi_2(\beta)}|2^{1-|\pi(\beta)|/2}\right], \tag{3.6}$$

and hence this expression is at least δ. We use this to establish good strategies for P_1 and P_2. We first establish that some parts of the given sum are small. We have the following result from [14, Lemma 6.9]

$$E_U[|\pi(\beta)|^{-1}] \le |\beta|^{-c}, \tag{3.7}$$

where c is a constant and in fact $c = \frac{1}{35}$ is possible. Note that the expectation is taken only over U and is true for any W.

Let $S_\delta = (4(6+2\log\delta^{-1})/\delta)^{1/c}$ and consider any β of size at least S_δ. Since $E[|\pi(\beta)|^{-1}] \le \delta/4(6+2\log\delta^{-1})^{-1}$, we conclude that the probability that $|\pi(\beta)| \le (6+2\log\delta^{-1})$ is upper bounded by $\delta/4$. Thus for any β of size at least S_δ we have

$$E_U[2^{1-|\pi(\beta)|/2}] \le Pr[|\pi(\beta)| \le (6+2\log\delta^{-1})] + 2^{2+\log\delta^{-1}} \le \frac{\delta}{4}+\frac{\delta}{4}=\frac{\delta}{2}$$

and hence discarding terms with $|\beta| \ge S_\delta$ in (3.6) still keeps a sum of expected value at least $\delta/2$.

Furthermore since $\sum_\beta \hat{B}_\beta^2 = 1$ we can discard any term with $|\hat{A}_{\pi_2(\beta)}| \le \delta/4$ and not reduce the sum by more than $\delta/4$. We conclude that the sum which is the right hand side of (3.6) is at least $\delta/4$ even if we restrict summation to β of size at most S_δ and such that $|\hat{A}_{\pi_2(\beta)}| \ge \delta/4$.

Now consider the following strategy for the provers P_1 and P_2. On receiving W, P_1 chooses β with probability $\hat{B}_\beta^2$ and returns a random $y \in \beta$. Similarly on receiving a U, P_2 chooses α with probability $\hat{A}_\alpha^2$ and returns a random $x \in \alpha$. We note that since A, B are folded over true, by Lemma 2.6, the sets α and β selected by the provers are always nonempty. Also, since B is conditioned upon φ_W, by Lemma 2.7, every $y \in \beta$ satisfies the formula φ_W. The success-probability of the given strategy is at least

$$E_{U,W}[\sum_{\beta}\hat{B}_{\beta}^{2}\hat{A}_{\pi_2(\beta)}^{2}|\beta|^{-1}]. \tag{3.8}$$

If we restrict summation to $|\beta| \le S_\delta$ and $|\hat{A}_{\pi_2(\beta)}| \ge \delta/4$, (3.8) is at least

$$S_\delta^{-1}\delta/4\; E_{U,W}\left[\sum_{\beta;|\beta|\le S_\delta,|\hat{A}_{\pi_2(\beta)}|\ge\delta/4}\hat{B}_{\beta}^{2}|\hat{A}_{\pi_2(\beta)}|\right]$$

and, by the above reasoning, this expected value is at least $\delta/4$ and we get a lower bound $S_\delta^{-1}(\delta/4)^2$ for the success probability of the provers. This completes the proof of Lemma 3.2. □

The basic test reads 5 bits (b_1,b_2,b_3,b_4,b_5) of the proof and checks whether $b_1b_2b_3(b_4 \wedge b_5) = 1$ which is same as $b_1 = b_2 \oplus b_3 \oplus (b_4 \wedge b_5)$ in $\{0,1\}$ notation. Theorem 1.2 now follows by a standard procedure of replacing the bits in the proof by variables and asking for a proof that maximizes the acceptance probability.

3.2 Iterated tests

We now extend our basic test in a query efficient way. We pick one set U and on it we pick k functions $(f_i)_{i=1}^k$ and k functions $(f'_j)_{j=1}^k$ and k sets $(W_l)_{l=1}^k$ each with its pair of functions (g_l, g'_l). Each W_l is picked uniformly from the set of possible companion in 2PP(u) to the already picked U. Thus for each l, (U, W_l) appears with the same probability as (U, W) in 2PP(u). Note that W_l is not independent of $W_{l'}$ for $l \neq l'$ as they are companions of the same U.

We perform the basic test for a certain set of quadruples (f_i, f'_j, g_l, g'_l). We give strong analyses in two cases each utilizing k^2 quadruples. One is given by the constraint $i = j$ and is analyzed very much as Samorodnitsky and Trevisan [12] analyzed their tests. We call it the "complete bipartite PCP".

The other set of k^2 quadruples is given by all triples (i, j, l) such that $i + j + l = 0 \bmod k$. The key property of this set of triples is that any two different triples have at most one coordinate in common. Hence we call it the "almost disjoint sets PCP". This analysis, done in the style of Håstad and Wigderson [15], is substantially simpler and hence we give this proof first.

In either case we get a test that reads $4k + k^2$ bits, has perfect completeness and soundness only marginally higher than 2^{-k^2}. Theorem 1.1 can therefore be obtained either form Theorem 3.3 below which analyze the almost disjoint sets PCP or Theorem 3.4 which analyzes the complete bipartite test.

3.2.1 The almost disjoint sets PCP

We first define the test which is an iteration of the basic test studied in the last section. The test depends on the parameter u used in 2PP(u) but we keep this dependence implicit to simplify notation.

k-iterated almost disjoint sets PCP

1. V chooses U as in 2PP(u).

2. V chooses independently k sets $(W_l)_{l=1}^k$, that can appear with U in 2PP(u). Each W_l is chosen with the distribution induced by 2PP(u), i.e., the distribution of the pair U, W_l is the same as the distribution of U, W in 2PP(u).

3. V chooses $2k$ functions $(f_i)_{i=1}^k$ and $(f'_j)_{j=1}^k$ on U uniformly at random.

4. For each l, $1 \le l \le k$, V chooses two functions g_l and g'_l on W_l uniformly at random.

5. For each triple i, j, l such that $i + j + l \equiv 0 \bmod k$ define a function h_{ijl} by setting for each $y \in \{-1, 1\}^{W_l}$, $h_{ijl}(y) = g_l(y) f_i(\pi(y))(g'_l(y) \wedge f'_j(\pi(y)))$.

6. V accepts iff $B_{W_l}(h_{ijl}) = B_{W_l}(g_l) A_U(f_i)(B_{W_l}(g'_l) \wedge A_U(f'_j))$ for all $i + j + l \equiv 0 \bmod k$.

We have the following theorem.

Theorem 3.3. *The k-iterated almost disjoint sets test has completeness* 1 *and soundness* $2^{-k^2} + d_c^{\Omega(u)}$, *where* d_c *is the constant from Theorem 2.1 and u is the parameter of the underlying 2-prover protocol.*

Proof. The completeness follows from that of the basic test and we need to analyze the soundness. For readability let us replace A_U by A and B_{W_l} by B_l. Let Z_0 denote the set of all triples (i,j,l) such that $i+j+l \equiv 0 (\bmod k)$.

Let $\mathrm{Acc}(i,j,l)$ be a variable that indicates whether the test given by the triple (i,j,l) accepts, taking the value 1 if it does and -1 otherwise. Clearly

$$\mathrm{Acc}(i,j,l) = B_l(h_{ijl})B_l(g_l)A(f_i)(A(f'_j) \wedge B_l(g'_l)).$$

Consider

$$\prod_{(i,j,l)\in Z_0} \frac{1+\mathrm{Acc}(i,j,l)}{2} = 2^{-k^2} \sum_{S\subseteq Z_0} \prod_{(i,j,l)\in S} \mathrm{Acc}(i,j,l). \tag{3.9}$$

This number equals 1 if the test accepts and is 0 otherwise and thus its expected value is the probability that the test accepts. The term in the right hand side sum with $S=\emptyset$ equals 1 and to establish the theorem it is sufficient to establish that any other term is bounded by $d_c^{\Omega(u)}$. Let Π_S be the term corresponding to $S \neq \emptyset$ and let T_S be the expectation of Π_S. We go on to establish strategies for P_1 and P_2 which makes the verifier in $2\mathrm{PP}(u)$ accept with probability $|T_S|^{O(1)}$. This is clearly sufficient to establish the theorem.

Suppose without loss of generality that $(k,k,k) \in S$ and let us fix the values of f_i, $i \neq k$, f'_j, $j \neq k$ and (W_l, g_l, g'_l) for $l \neq k$ in such a way that the conditional expectation of Π_S remains at least T_S. As the sets in Z_0 only intersect in one point we can, up to a factor ±1, write Π_S as

$$\mathrm{Acc}(k,k,k) \prod_{(k,j,l)\in S, j,l\neq k} \mathrm{Acc}(k,j,l) \prod_{(i,k,l)\in S, i,l\neq k} \mathrm{Acc}(i,k,l) \prod_{(i,j,k)\in S, i,j\neq k} \mathrm{Acc}(i,j,k) \tag{3.10}$$

as the rest of the variables are fixed. The three products of (3.10) can be written as $A^{(1)}(f_k)$, $A^{(2)}(f'_k)$ and $B^{(1)}(g_k, g'_k)$ respectively, for some Boolean functions $A^{(1)}$, $A^{(2)}$ and $B^{(1)}$.

Expanding the definition of $\mathrm{Acc}(k,k,k)$ and using $x \wedge y = \frac{1+x+y-xy}{2}$ for $A(f'_k) \wedge B_k(g'_k)$ we see that (3.10) can be written as the sum of four terms of the form

$$B_k(h_{kkk})A'(f_k)A''(f'_k)C(g_k, g'_k), \tag{3.11}$$

each with a coefficient $1/2$, for some Boolean functions A', A'' and C closely related to $A^{(1)}$, $A^{(2)}$ and $B^{(1)}$. To be more precise

$$A'(f_k) = A(f_k)A^{(1)}(f_k),$$

$$A''(f'_k) = A^{(2)}(f'_k) \quad \text{or} \quad A''(f'_k) = A(f'_k)A^{(2)}(f'_k),$$

and

$$C(g_k, g'_k) = B_k(g_k)B^{(1)}(g_k, g'_k) \quad \text{or} \quad C(g_k, g'_k) = B_k(g_k)B_k(g'_k)B^{(1)}(g_k, g'_k).$$

We want to prove that if the expectation of (3.10) is large then the provers P_1 and P_2 in the two prover game can convince the verifier of that protocol to accept with high probability. To this end we use the tables in the given PCP to construct strategies for P_1 and P_2. We need to be slightly careful since not all derived tables can be used by a given prover as it might depend on information not available to this particular prover.

In the present situation the functions A' and A'' depend only on U and the fixations made and hence are available for player P_2 to design a strategy. B_k is the original long code on W_k and hence is useful for extracting a strategy for P_1. Finally C is a function that depends on both U and W_k and as this is not fully known to either P_1 or P_2, C is not useful for designing strategies.

Since we only have one remaining object of each type, let us for readability discard the index replacing f_k by f, W_k by W, etc.

We now want to compute the expected value of (3.11) over random choices of f, f', g and g'. Expanding all factors except $A''(f')$ by the Fourier transform we get

$$\sum_{\alpha,\beta,\gamma,\gamma'} \hat{A}'_\alpha \hat{B}_\beta \hat{C}_{\gamma,\gamma'} E_{f,f',g,g'}\left[\chi_\alpha(f)\chi_\beta(gf(f'\wedge g'))\chi_\gamma(g)\chi_{\gamma'}(g')A''(f')\right]. \tag{3.12}$$

Taking the expectation over f we see, using Lemma 2.3, that any term with $\alpha \neq \pi_2(\beta)$ vanishes while if we have equality the expectation is 1. Similarly, considering the expectation over g, we see that only terms with $\beta = \gamma$ give a nonzero contribution. Finally, fixing f' and considering expectation over g', we see that only terms with $\gamma' = \beta \cap \pi^{-1}(f'^{-1}(-1))$ remain nonzero.

This implies that (3.12) reduces to

$$E_{U,W,f'}\left[\sum_\beta \hat{A}'_{\pi_2(\beta)} \hat{B}_\beta \hat{C}_{\beta,\beta\cap\pi^{-1}(f'^{-1}(-1))} A''(f')\right] \tag{3.13}$$

and, fixing U and W, let us estimate

$$E_{f'}\left[\sum_\beta \hat{A}'_{\pi_2(\beta)} \hat{B}_\beta \hat{C}_{\beta,\beta\cap\pi^{-1}(f'^{-1}(-1))} A''(f')\right]. \tag{3.14}$$

Towards this end we have

$$\begin{aligned} \mid E_{f'}[\hat{C}_{\beta,\beta\cap\pi^{-1}(f'^{-1}(-1))}A''(f')] \mid \quad &\leq \quad E_{f'}[|\hat{C}_{\beta,\beta\cap\pi^{-1}(f'^{-1}(-1))}|] \leq \\ 2^{-|\pi(\beta)|}\sum_{\alpha'\subseteq\pi(\beta)} |\hat{C}_{\beta,\beta\cap\pi^{-1}(\alpha')}| \quad &\leq \quad 2^{-|\pi(\beta)|/2}\left(\sum_{\alpha'\subseteq\pi(\beta)} \hat{C}^2_{\beta,\beta\cap\pi^{-1}(\alpha')}\right)^{1/2}. \end{aligned} \tag{3.15}$$

Substituting this estimate into (3.14) we get the upper estimate

$$\sum_\beta |\hat{A}'_{\pi_2(\beta)}\hat{B}_\beta| 2^{-|\pi(\beta)|/2}\left(\sum_{\alpha'\subseteq\pi(\beta)} \hat{C}^2_{\beta,\beta\cap\pi^{-1}(\alpha')}\right)^{1/2} \tag{3.16}$$

and applying the Cauchy-Schwartz inequality over β this is bounded by

$$\left(\sum_\beta \hat{B}^2_\beta \hat{A}'^2_{\pi_2(\beta)} 2^{-|\pi(\beta)|}\right)^{1/2}\left(\sum_{\beta,\beta_1}\hat{C}^2_{\beta,\beta_1}\right)^{1/2} \leq \left(\sum_\beta \hat{B}^2_\beta \hat{A}'^2_{\pi_2(\beta)} 2^{-|\pi(\beta)|}\right)^{1/2}, \tag{3.17}$$

which is our final upper bound for the absolute value of the expectation of Π_S when U and W are fixed. As $E[X^2] \geq E[X]^2$ we have

$$E_{U,W}\left[\sum_{\beta} \hat{B}_{\beta}^2 \hat{A}'^2_{\pi_2(\beta)} 2^{-|\pi(\beta)|}\right] \geq E_{U,W}\left[|E_{f,f',g,g'}[\Pi_S]|^2\right] \geq E_{U,W}\left[|E_{f,f',g,g'}[\Pi_S]|\right]^2 \geq E[\Pi_S]^2 \geq T_S^2.$$

The rest of the proof now follows along the same lines as end of the proof for the basic test. In that proof we had established that the right hand side of (3.6) was large and used this to derive strategies for the provers. We now have proved that a very similar sum is large. The fact that we have replaced $\hat{A}_{\pi_2(\beta)}$ by $\hat{A}'^2_{\pi_2(\beta)}$ is only to our advantage. As A' is a derived table we cannot make sure that it is folded over true and thus when P_2 picks α with probability $\hat{A}'^2_{\alpha}$ the set α might be empty. In this case P_2 might return any assignment and we assume that the verifier rejects in this case. This does not disturb the analysis as B is folded over true and hence $|\beta|$ is odd which implies that $\pi_2(\beta)$ is nonempty. □

3.2.2 The bipartite graph test

In this section we study the following test.

k- iterated bipartite graph PCP

1. V chooses U as in 2PP(u).

2. V chooses independently k sets $(W_l)_{l=1}^k$, that can appear with U in 2PP(u). Each W_l is chosen with the distribution induced by 2PP(u), i.e., the distribution of the pair U, W_l is the same as the distribution of U, W in 2PP(u).

3. V chooses $2k$ functions $(f_i)_{i=1}^k$ and $(f_i')_{i=1}^k$ on U uniformly at random.

4. For each l, $1 \leq l \leq k$, V chooses two functions g_l and g_l' on W_l uniformly at random.

5. For each pair i,l define a function h_{il} by setting for each $y \in \{-1,1\}^{W_l}$, $h_{il}(y) = g_l(y) f_i(\pi(y))(g_l'(y) \wedge f_i'(\pi(y)))$.

6. V accepts iff $B_{W_l}(h_{il}) = B_{W_l}(g_l) A_U(f_i)(B_{W_l}(g_l') \wedge A_U(f_i'))$ for all $1 \leq i,l \leq k$.

We have the following theorem.

Theorem 3.4. *The bipartite graph test has completeness* 1 *and soundness* $2^{-k^2} + d_c^{\Omega(u)}$.

Proof. The completeness is again not difficult and we leave it the reader to verify that indeed V always accepts a correct proof for a correct statement.

In the analysis of the soundness let us use notation similar to the one used in the previous proof, e.g., writing B_l instead of B_{W_l} and A instead of A_U. Also define

$$\mathrm{Acc}(i,l) = B_l(h_{il}) B_l(g_l) A(f_i)(A(f_i') \wedge B_l(g_l')),$$

which is 1 if the test involving h_{il} accepts and -1 if the test fails. Now we want to calculate the expected value of

$$\prod_{(i,l)\in[k]\times[k]} \frac{1+\mathrm{Acc}(i,l)}{2} = 2^{-k^2} \sum_{S\subseteq[k]\times[k]} \prod_{(i,l)\in S} \mathrm{Acc}(i,l). \tag{3.18}$$

Let T_S be the expectation of the product for S and the goal is again to, for any nonempty set S, give a prover-strategy with success rate $|T_S|^{O(1)}$. We start by, as already done in [12], reducing to the case of special S and let T_{2d} be the result when S is the edge set of the complete bipartite graph on $[2]\times[d]$.

Lemma 3.5. *[12] For any nonempty S, there is an integer d such that* $|T_S| \le |T_{2d}|^{1/2}$.

Proof. As all coordinates are treated symmetrically me may, without loss of generality, assume that $(1,1)\in S$ and that $(1,2),\ldots(1,d)$ are the other vertices in S connected to 1. Let us divide our random choice of $(f_i, f_i', g_l, g_l')_{i,l=1,..,k}$ into X given by choice of (f_1, f_1'), and Y given by choice of the rest. Let S_1 be the subset of S given by $(1,1),(1,2)\ldots(1,d)$. Then

$$\begin{aligned} E_{X,Y}[\prod_{(i,l)\in S} \mathrm{Acc}(i,l)] = E_{X,Y}[\prod_{l=1}^{d} \mathrm{Acc}(1,l)\cdot \prod_{(i,l)\in S\setminus S_1} \mathrm{Acc}(i,l)] = \\ E_{X,Y}[F(X,Y)G(Y)] = E_Y[E_X[F(X,Y)]G(Y)] \end{aligned}$$

for some functions F and G with values in $\{-1,1\}$. Now applying Cauchy-Schwartz inequality this can be bounded by

$$\begin{aligned} \sqrt{E_Y[(E_X[F(X,Y)])^2]}\sqrt{E_Y[G(Y)^2]} &\le \sqrt{E_Y[(E_X[F(X,Y)])^2]} = \\ \sqrt{E_Y[E_{X_1}[F(X_1,Y)]\cdot E_{X_2}[F(X_2,Y)]]} &= \sqrt{E_{X_1,X_2,Y}[F(X_1,Y)\cdot F(X_2,Y)]} \end{aligned}$$

where X_1, X_2 are identically distributed as X and are independent. The proof is completed by the observation that $F(X_1,Y)\cdot F(X_2,Y)$ is equal to $\prod_{l=1}^{d}\mathrm{Acc}(1,l)\cdot\prod_{l=1}^{d}\mathrm{Acc}(2,l)$, which is exactly the same as $T_{S'}$ where S' is a complete bipartite graph on $[2]\times[d]$. □

Thus it is sufficient to find a good strategy based on $|T_{2d}|$ being large. Using the definition of Acc and cancelling the factors $B_l(g_l)$ that appears exactly twice, we have

$$T_{2d} = E\left[\prod_{l=1}^{d} B_l(h_{1l})B_l(h_{2l})A(f_1)A(f_2)(A(f_1')A(f_2')\wedge B_l(g_l'))\right] \tag{3.19}$$

The function g_l affects T_{2d} only through h_{1l} and h_{2l} and replacing $B_l(h_{1l})$ and $B_l(h_{2l})$ by their Fourier expansions we see that

$$\begin{aligned} E_{g_l}[B_l(h_{1l})B_l(h_{2l})] &= \sum_{\beta_1,\beta_2} \hat{B}_{l,\beta_1}\hat{B}_{l,\beta_2} E_{g_l}[\chi_{\beta_1}(g_l f_1(g_l'\wedge f_1'))\chi_{\beta_2}(g_l f_2(g_l'\wedge f_2'))] = \\ & \sum_{\beta} \hat{B}_{l,\beta}^2 \chi_\beta(f_1 f_2)\chi_\beta(f_1' f_2'\wedge g_l'). \end{aligned} \tag{3.20}$$

Substituting this into (3.19) we get

$$E\left[\prod_{l=1}^{d}\left(\sum_{\beta_l}\hat{B}^2_{l,\beta_l}\chi_{\beta_l}(f_1f_2)\chi_{\beta_l}(f'_1f'_2\wedge g'_l)\right)A(f_1)A(f_2)((A(f'_1)A(f'_2))\wedge B_l(g'_l))\right] \tag{3.21}$$

Let us now consider the expectation over f_1 and f_2. If d is even then the dependence of (3.21) on f_1 and f_2 is of the form

$$\prod_{l=1}^{d}\chi_{\beta_l}(f_1f_2)$$

which has expected value 0 unless $\oplus_j\pi_2(\beta_l)=\emptyset$ while the expectation is 1 if we have equality.

If d is odd, then the dependence of f_1 and f_2 is of the form

$$A(f_1)A(f_2)\prod_{l=1}^{d}\chi_{\beta_l}(f_1f_2).$$

Replacing $A(f_1)$ and $A(f_2)$ by their Fourier expansions we see that the expectation of this with respect to f_1 and f_2 equals $\hat{A}^2_\alpha$ where

$$\alpha=\oplus_l\pi_2(\beta_l).$$

Now let us turn to analyzing the rest of (3.21). First note that

$$\prod_{l=1}^{d}(A(f'_1)A(f'_2)\wedge B_l(g'_l))=(A(f'_1)A(f'_2)\wedge\prod_{l=1}^{d}B_l(g'_l)).$$

We have $(x\wedge y)=\frac{1+x+y-xy}{2}$ and we are now ready to consider the expectation over f'_1 and f'_2 and g'_l. We have expressions of the form

$$(A(f'_1)A(f'_2))^a\prod_{j=l}^{d}\chi_{\beta_l}(f'_1f'_2\wedge g'_l)(\prod_{l=1}^{d}B_l(g'_l))^b, \tag{3.22}$$

for $a,b\in\{0,1\}$. Now, view

$$C(g'_1,g'_2\dots g'_d)=(\prod_{l=1}^{d}B_l(g'_l))^b$$

as a Boolean function with Fourier coefficients $\hat{C}_{\gamma_1,\gamma_2,\dots,\gamma_d}$, and thus (3.22) equals

$$\sum_{\gamma_1,\gamma_2,\dots\gamma_l}(A(f'_1)A(f'_2))^a\hat{C}_{\gamma_1,\gamma_2,\dots,\gamma_d}\prod_{j=l}^{d}\chi_{\beta_l}(f'_1f'_2\wedge g'_l)\chi_{\gamma_l}(g'_l). \tag{3.23}$$

Let $\alpha'=\cup_{l=1}^{d}\pi(\beta_l)$. For a fixed choice of $f'_1f'_2=f'$ we get a nonzero expected value over $(g'_l)_{l=1}^{d}$ iff $\gamma_l=\beta_l\cap\pi^{-1}(f'^{-1}(-1))$ for all l, giving a unique non-zero term. Defining $\gamma_l^{\vec{\beta},f'}$ to be this value we get

$$\left|E_{f'_1,f'_2,g'_1,g'_2,\dots g'_d}\left[(A(f'_1)A(f'_2))^a\prod_{j=l}^{d}\chi_{\beta_l}(f'_1f'_2\wedge g'_l)(\prod_{l=1}^{d}B_l(g'_l))^b\right]\right| \quad\leq \tag{3.24}$$

$$\left|E_{f'_1,f'_2}\left[\hat{C}_{\gamma_1^{\vec{\beta},f'},\gamma_2^{\vec{\beta},f'},\dots,\gamma_d^{\vec{\beta},f'}}\right]\right| \quad\leq\quad 2^{-|\alpha'|/2}, \tag{3.25}$$

where the last inequality follows from Cauchy-Schwartz's inequality using a similar calculation to that in (3.5). This means that in the case when d is even we get the upper estimate

$$\sum_{\oplus_l \pi_2(\beta_l)=\emptyset} \prod_{l=1}^{d} \hat{B}^2_{l,\beta_l} 2^{-|\alpha'|/2} \tag{3.26}$$

for $|T_{2d}|$ while in the case when d is odd we get

$$\sum \hat{A}^2_{\oplus_l \pi_2(\beta_l)} \prod_{l=1}^{d} \hat{B}^2_{l,\beta_l} 2^{-|\alpha'|/2}, \tag{3.27}$$

where in both cases we have $\alpha' = \cup_{l=1}^{d} \pi(\beta_l)$.

Strategies for the provers can now be defined as follows. P_1 upon receiving W, picks β with probability $\hat{B}^2_\beta$ and returns a random $y \in \beta$. P_2 upon receiving U picks $d-1$ random W_l, $l = 2\ldots d$ and picks $\beta_2, \ldots \beta_d$ with probability $\prod_{l=2}^{d} \hat{B}^2_{l,\beta_l}$ and computes $\alpha'' = \oplus_{l=2}^{d} \pi_2(\beta_l)$. If d is even P_2 returns a random $x \in \alpha''$. If d is odd P_2 also picks α with probability $\hat{A}^2_\alpha$ and returns a random element in $\alpha'' \oplus \alpha$. Note by folding, in both cases the defined set is of odd cardinality and hence it is not empty.

The probability of success is, in the case of even d, at least

$$\sum_{\oplus_l \pi_2(\beta_l)=\emptyset} \prod_{l=1}^{d} \hat{B}^2_{l,\beta_l} (\sum |\beta_l|)^{-1} \tag{3.28}$$

and in the case of odd d it as at least

$$\sum \hat{A}^2_{\oplus_l \pi_2(\beta_l)} \prod_{l=1}^{d} \hat{B}^2_{l,\beta_l} (\sum |\beta_l|)^{-1}. \tag{3.29}$$

Using (3.7) these probabilities can be related to expressions (3.26) and (3.27) in a way similar to the basic proof case. We omit the details. The result is that the verifier in 2PP(u) accepts with probability $|T_{2d}|^{O(1)}$ and the theorem follows. □

3.2.3 Adaptive tests

In this section we prove Theorem 1.4 by defining a suitable adaptive test. The theorem then follows from analyzing the completeness, which is done in Lemma 3.6 and the soundness which is done in Lemma 3.7 Guruswami et al. [7] give an adaptive test reading three bits that has perfect completeness and soundness $\frac{1}{2} + \varepsilon$ for any $\varepsilon > 0$. The non-adaptive version of this test has the same parameters except that it reads 4 bits. The natural iterated test based on this test reads $2k + k^2$ bits in the adaptive setting and $2k + 2k^2$ bits in the non-adaptive setting. It has perfect completeness and it turns out that soundness is essentially 2^{-k^2} also for this test.

Thus its parameters, when adaptive, are the same as those of the test of Samorodnitsky and Trevisan while achieving perfect completeness. As sketched in [8], this test can be designed and analyzed with the same basic two-prover protocol as the previous tests but the construction turns out to be technically simpler if we modify the two-prover protocol. We do this to obtain the property called "smoothness" in [10]. We need that for two different answers by P_1, with high probability the answers by P_2 causing acceptance are also different. This is achieved by sending a large number of identical clauses to both provers.

u-parallel two-prover protocol with T factor extra clauses, 2PPe$(\mathbf{u},\mathbf{T})$

1. V chooses Tu clauses $(C_{k_i})_{i=1}^{Tu}$ uniformly at random. Then he randomly selects u clauses $(C_{j_i})_{i=1}^{u}$ out of these Tu clauses and randomly selects a variable x_{j_i} from each clauses C_{j_i}. He sends $(k_i)_{i=1}^{Tu}$ to prover P_1 and to prover P_2, the u chosen variables $(x_{j_i})_{i=1}^{u}$ together with the $(T-1)u$ clauses not selected.

2. V receives values for u chosen variables $(x_{j_i})_{i=1}^{u}$ from P_2 as well as $3(T-1)u$ values for the variables in the clauses sent to P_2. V also receives $3Tu$ values from P_1 to the variables in the clauses sent to P_1. V accepts if no two values are inconsistent and all the picked clauses are satisfied.

We again call the sets of variables sent to the two provers U and W, respectively. Note that this time U is of size $u(3T-2)$ and W is of size $3uT$ while as before we have $U \subset W$. Note also that for each fixed set of $(T-1)u$ clauses sent to both players, we have an instance of the 2PP(u). This implies that the soundness of 2PPe(u,T) is at most that of 2PP(u) and in particular it is upper bounded by d_c^u.

We now describe the PCP. It depends on the parameters u and T but has also additional parameters k and ε. For notational convenience we suppress the former.

k-iterated non-adaptive PCP of bias ε

1. V chooses U as in 2PPe(u,T).

2. V chooses independently k sets $(W_j)_{j=1}^{k}$, that can appear with U in 2PPe(u,T). Each W_j is chosen with the distribution induced by 2PPe(u,T), i.e., the distribution of the pair U,W_j is the same as the distribution of U,W in 2PPe(u,T).

3. V chooses k functions $(f_i)_{i=1}^{k}$ on U uniformly at random and reads the bits $A_U(f_i)$.

4. For each j, $1 \le j \le k$, V chooses a function g_j on W_j uniformly at random and reads the bits $B_{W_j}(g_j)$.

5. For each pair i,j define a function h_{ij} by setting, independently, for each $y \in \{-1,1\}^{W_j}$, $h_{ij}(y) = -1$ with probability $1-\varepsilon$ and otherwise $h_{ij}(y) = 1$.

6. For each pair i,j, if $A(f_i) = 1$, V checks that $B_j(g_j(f_i \wedge h_{ij})) = B_j(g_j)$ and otherwise V checks that $B_j(g_j(-f_i \wedge h_{ij})) = B_j(g_j)$.

7. V accepts if all tests accept.

Completeness is straightforward.

Lemma 3.6. *The adaptive k-iterated test with bias ε, accepts with probability 1, i.e, it has perfect completeness.*

Proof. Fix an i and a j. Suppose that we have a correct proof of a correct statement based on the global assignment z. If $A(f_i) = 1$ then $f_i(\pi^U(z)) = 1$ and we have

$$B_j(g_j(f_i \wedge h_{ij})) = g_j(\pi^W(z))(f_i(\pi^U(z)) \wedge h_{ij}(\pi^W(z))) = g_j(\pi^W(z)) = B_j(g_j).$$

The case $A(f_i) = -1$ is similar. □

We next turn to soundness.

Lemma 3.7. *Suppose that $T \geq \varepsilon^{-5}$ and we are given a proof that makes the verifier in the adaptive iterated test with parameter ε accept with probability $2^{-k^2} + 2\delta$ where $\delta > 6\varepsilon$. Then we can find strategies for P_1 and P_2 in 2PPe(u,T) that makes the verifier of that protocol accept with probability at least $\varepsilon^2(\delta - 6\varepsilon)^2/2$.*

Proof. The proof follows along the same lines as the result for the protocol with $k = 1$ given in [7] which in turn is based on the proof that 3SAT is inapproximable for satisfiable instances in [14].

Let

$$\mathrm{Acc}(i,j) = \frac{1}{2}\left((1 + A(f_i))B_j(g_j)B_j(g_j(f_i \wedge h_{ij})) + (1 - A(f_i))B_j(g_j)B_j(g_j(-f_i \wedge h_{ij}))\right),$$

which is 1 if the test given by (i,j) accepts and -1 otherwise. We have an expansion like (3.18) and by the assumption of the lemma implies that we have a nonempty S such that

$$E\left[\prod_{(i,j)\in S} \mathrm{Acc}(i,j)\right] \geq 2\delta. \tag{3.30}$$

As all coordinates are symmetric we may assume that $(1,1) \in S$. Now fix the values of g_j and f_i for $i,j \geq 2$ and h_{ij} for $(i,j) \neq (1,1)$ to any constants without decreasing the expected value obtaining

$$E_{U,W_1,f_1,g_1,h_{11}}\left[\mathrm{Acc}(1,1)A^{(1)}(f_1)B^{(1)}(g_1)\right] \geq 2\delta \tag{3.31}$$

for some Boolean functions $A^{(1)}$ and $B^{(1)}$. Using the expression for Acc$(1,1)$ we get an expression of the form

$$A'(f_1)B(g_1(f_1 \wedge h_{11}))C(g_1) \tag{3.32}$$

or

$$A'(f_1)B(g_1(-f_1 \wedge h_{11}))C(g_1) \tag{3.33}$$

whose expectation over the choice of U, f_1, W_1, g_1 and h_{11} is at least δ. Here A', B and C are Boolean functions where B is the original B_1 and A' is a function only depending on U. Since f is chosen with the same distribution as $-f$ we might as well study (3.32) and let us drop the subscripts for readability. Replacing each function by its Fourier expansion, we get that the expectation of (3.32) equals

$$E_{U,W,f,g,h}\left[\sum_{\alpha,\beta,\gamma}\hat{A}'_{\alpha}\hat{B}_{\beta}\hat{C}_{\gamma}\chi_{\alpha}(f)\chi_{\beta}(g(f\wedge h))\chi_{\gamma}(g)\right]. \tag{3.34}$$

Taking expectation over g we see that terms with $\beta\neq\gamma$ have expectation 0 and thus (3.34) equals

$$E_{U,W}\left[\sum_{\alpha,\beta}\hat{A}'_{\alpha}\hat{B}_{\beta}\hat{C}_{\beta}E_{f,h}\left[\chi_{\alpha}(f)\chi_{\beta}((f\wedge h))\right]\right]. \tag{3.35}$$

If $\alpha\not\subseteq\pi(\beta)$ the expectation over f yields 0 and thus we need to study

$$E_{f,h}[\chi_{\alpha}(f)\chi_{\beta}(f\wedge h)] \tag{3.36}$$

where $\alpha\subseteq\pi(\beta)$. Using the definition of the characters (3.36) equals

$$E_{f,h}\left[\prod_{x\in\alpha}\left(f(x)\prod_{y\in\beta\cap\pi^{-1}(x)}(f(x)\wedge h(y))\right)\prod_{x\in\pi(\beta)\setminus\alpha}\left(\prod_{y\in\beta\cap\pi^{-1}(x)}(f(x)\wedge h(y))\right)\right] \tag{3.37}$$

and as the different x behave independently we can analyze each factor independently. We have $f(x)=1$ with probability $1/2$ and in this case

$$\prod_{y\in\beta\cap\pi^{-1}(x)}(f(x)\wedge h(y)))=1,$$

while while if $f(x)=-1$, it has expectation over h that equals $(2\varepsilon-1)^{s_x}$ where $s_x=|\pi^{-1}(x)\cap\beta|$. We conclude that the expectation of (3.37) equals

$$\prod_{x\in\alpha\cap\pi(\beta)}(\frac{1}{2}(1-(2\varepsilon-1)^{s_x}))\prod_{x\in\pi(\beta)\setminus\alpha}(\frac{1}{2}(1+(2\varepsilon-1)^{s_x})),$$

and defining this expression to be $p(\alpha,\beta)$, we conclude that (3.35) equals

$$E_{U,W}\left[\sum_{\beta,\alpha\subseteq\pi(\beta)}\hat{A}'_{\alpha}\hat{B}_{\beta}\hat{C}_{\beta}p(\alpha,\beta)\right]. \tag{3.38}$$

By assumption this expectation is at least δ and we need to design strategies for P_1 and P_2.

The strategies of the two provers are the standard strategies. i.e., P_2 chooses an α with probability $\hat{A}'^2_{\alpha}$ and returns a random $x\in\alpha$. Similarly P_1 chooses a random β with probability $\hat{B}^2_{\beta}$ and returns a random $y\in\beta$. Again A' cannot be assumed to be folded as it is a derived table. If α is the empty set we do not care what P_2 does and we assume in the analysis that the verifier rejects. The table B, on the other hand, is the original table and hence β is nonempty and any $y \in \beta$ satisfies the selected clauses. We conclude that the strategy of the provers is successful with probability at least

$$E_{U,W}\left[\sum_{\beta,\emptyset\neq\alpha\subseteq\pi(\beta)} \hat{A}'^2_\alpha \hat{B}^2_\beta |\beta|^{-1}\right]. \tag{3.39}$$

We need to prove that this is large based on (3.38) being at least δ.

First note that

$$\sum_\beta |\hat{B}_\beta \hat{C}_\beta| \le \left(\sum_\beta \hat{B}^2_\beta\right)^{1/2} \left(\sum_\beta \hat{C}^2_\beta\right)^{1/2} \le 1, \tag{3.40}$$

and the quantity that multiplies $\hat{B}_\beta \hat{C}_\beta$ in (3.38) satisfies

$$\begin{aligned} |\sum_{\alpha\subseteq\pi(\beta)} \hat{A}'_\alpha p(\alpha,\beta)| \quad &\le \quad \left(\sum_{\alpha\subseteq\pi(\beta)} \hat{A}'^2_\alpha\right)^{1/2} \left(\sum_{\alpha\subseteq\pi(\beta)} p^2(\alpha,\beta)\right)^{1/2} \\ \le \left(\sum_{\alpha\subseteq\pi(\beta)} p^2(\alpha,\beta)\right)^{1/2} \quad &\le \quad (1-\varepsilon)^{|\pi(\beta)|/2}. \end{aligned} \tag{3.41}$$

To see the last inequality in (3.41) note that the sum equals

$$\prod_{x\in\pi(\beta)} \left((\frac{1}{2}(1-(2\varepsilon-1)^{s_x}))^2 + (\frac{1}{2}(1+(2\varepsilon-1)^{s_x}))^2 \right). \tag{3.42}$$

The factor corresponding to x in (3.42) is of the form a^2+b^2 where $|a|+|b|=1$ and $\max(|a|,|b|) \le 1-\varepsilon$, and hence it is bounded by $(1-\varepsilon)$ and this gives the bound.

Our redesigned two-prover protocol enables us to control the size of projections nicely.

Lemma 3.8. *For any fixed W and β we have*

$$\Pr_U[|\pi(\beta)| < |\beta|] < \frac{|\beta|^2}{2T}. \tag{3.43}$$

Proof. For the event in (3.43) to happen there must be two different elements of β that project to the same element. There are at most $|\beta|^2/2$ pairs and the probability that any pair project to the same element is at most $1/T$. This follows since two different elements differ in at least one coordinate and the probability that a given coordinate does not appear in U is bounded above by $1/T$. The lemma follows from the union bound. □

Let us return to (3.38) and consider the terms corresponding to a fixed β. If $|\beta| \ge 2\varepsilon^{-2}$ then using Lemma 3.8, we see, as $T \ge \varepsilon^{-5}$, that except with probability 2ε we have $|\pi(\beta)| \ge 2\varepsilon^{-2}$ in which case (3.41) is bounded by

$$(1-\varepsilon)^{\varepsilon^{-2}} \le e^{\varepsilon^{-1}} \le \varepsilon$$

We conclude that

$$E_{U,W}\left[\sum_{|\beta|\ge 2\varepsilon^{-2},\alpha\subseteq\pi(\beta)} \hat{A}'_\alpha \hat{B}_\beta \hat{C}_\beta p(\alpha,\beta)\right] \le E_{U,W}\left[\sum_{|\beta|\ge 2\varepsilon^{-2}} \hat{B}_\beta \hat{C}_\beta (Pr[|\pi(\beta)| \le 2\varepsilon^{-1}] + \varepsilon)\right] \le 3\varepsilon. \tag{3.44}$$

It follows that

$$E_{U,W}\left[\sum_{|\beta|\le 2\varepsilon^{-2},\alpha\subseteq\pi(\beta)} \hat{A}'_\alpha \hat{B}_\beta \hat{C}_\beta p(\alpha,\beta)\right] \ge \delta - 3\varepsilon, \tag{3.45}$$

and we want to bound the contribution from $\alpha = \emptyset$. Note that if $|\beta| \le 2\varepsilon^{-2}$ then, by Lemma 3.8, except with probability 2ε each s_x is one. In this case

$$p(\emptyset,\beta) = \varepsilon^{|\beta|} \le \varepsilon$$

and we conclude that the total expectation of terms containing $\alpha = \emptyset$ is at most 3ε and hence we have

$$E_{U,W}\left[\sum_{|\beta|\le 2\varepsilon^{-2},\emptyset\ne\alpha\subseteq\pi(\beta)} \hat{A}'_\alpha \hat{B}_\beta \hat{C}_\beta p(\alpha,\beta)\right] \ge \delta - 6\varepsilon. \tag{3.46}$$

Returning to (3.39) we see that the provers are successful with with probability at least

$$\frac{\varepsilon^2}{2} E_{U,W}\left[\sum_{\beta,\emptyset\ne\alpha\subseteq\pi(\beta),|\beta|\le 2\varepsilon^{-2}} \hat{A}'^2_\alpha \hat{B}^2_\beta\right].$$

Now by the above reasoning we have the following chain of equalities, where all sums are over the set

$$\{\beta,\emptyset\ne\alpha\subseteq\pi(\beta),|\beta|\le 2\varepsilon^{-2}\}.$$

$$\begin{aligned}(\delta-6\varepsilon)^2 \le \left(E_{U,W}\left[\sum \hat{A}'_\alpha \hat{B}_\beta \hat{C}_\beta p(\alpha,\beta)\right]\right)^2 &\le E_{U,W}\left[\left(\sum \hat{A}'_\alpha \hat{B}_\beta \hat{C}_\beta p(\alpha,\beta)\right)^2\right] \le \\ E_{U,W}\left[\left(\sum \hat{A}'^2_\alpha \hat{B}^2_\beta\right)\left(\sum \hat{C}^2_\beta p^2(\alpha,\beta)\right)\right] &\le E_{U,W}\left[\sum \hat{A}'^2_\alpha \hat{B}^2_\beta\right],\end{aligned}$$

where the last inequality follows from

$$\sum_\beta \sum_{\alpha\subseteq\beta} \hat{C}^2_\beta p^2(\alpha,\beta) \le \sum_\beta \hat{C}^2_\beta \le 1.$$

where we again used the last inequality of (3.41). We conclude that the verifier in the two-prover protocol accepts with the given strategies with probability at least $\varepsilon^2(\delta-6\varepsilon)^2/2$ and the proof is complete. □

4 The case of larger domains

In this section we prove Theorem 1.5. This is done by a natural extension of the protocols from the previous sections. Before we present our protocols we give some definitions and recall some background results.

4.1 Background in the large domain case

Let Z_p denote the multiplicative group given by the p^{th} roots of unity. Let $\zeta = e^{2\pi i/p}$ be the basic p^{th} root of unity. To generalize the Boolean $\wedge$ we define an operation mult(,) as:

$$\text{mult}(\zeta^i, \zeta^j) = \zeta^{ij}.$$

We have the following useful lemma

Lemma 4.1. *For x and y being p^{th} roots of unity we have*

$$\text{mult}(x,y) = \frac{1}{p}\sum_{i=0}^{p-1}\sum_{j=0}^{p-1} x^i y^j \zeta^{-ij}.$$

Proof. Suppose $y = \zeta^{i_0}$. Fix i and consider the inner sum. For $i \neq i_0$ the value is 0 while for $i = i_0$ it is p. This implies that the total sum equals x^{i_0} which is in fact mult(x,y). □

We define long p codes as the natural extension of the long code. Positions are indexed by functions $f : \{-1,1\}^t \to Z_p$ and in the code for x this position takes value $f(x)$.

Let A be a table containing a value $A(f) \in Z_p$ for every function $f : \{-1,1\}^t \to Z_p$. We make the following definitions for such a table.

Definition 4.2. A table A is folded over true if $A(\zeta^a f) = \zeta^a A(f)$, for $0 \leq a \leq p-1$ and all f.

Definition 4.3. A table A respects exponentiation if $A(f^a) = A(f)^a$ for $0 \leq a \leq p-1$ and all f.

Definition 4.4. A table A is conditioned upon a function $h : \{-1,1\}^t \to \{1,\zeta\}$ (1 represents false and ζ represents true), if $A(f) = A(\text{mult}(f,h))$ for all f.

Now we briefly explain Fourier analysis of long p-codes. For every function $\alpha : \{-1,1\}^t \to GF(p)$, where $GF(p)$ is represented by $\{0,1,\dots p-1\}$, there is a character χ_α defined as

$$\chi_\alpha(f) = \prod_{x\in\{-1,1\}^t} f(x)^{\alpha(x)}$$

Note that α is a "function" rather than a "set" as in binary case and that the transform takes complex values. We denote by $N(\alpha)$ the set on which α takes nonzero values i.e.,

$$N(\alpha) = \{x|\alpha(x) \neq 0\}.$$

Every table A can be written as $A(f) = \sum_\alpha \hat{A}_\alpha \chi_\alpha(f)$ with $\sum_\alpha |\hat{A}_\alpha|^2 = 1$. We can assume that tables are folded or conditioned upon a given function by using appropriate access mechanisms. Following are easy consequences of folding and conditioning.

Lemma 4.5. *If A is folded over true and $\hat{A}_\alpha \neq 0$, then $\sum_{x\in\{-1,1\}^t} \alpha(x) = 1 \mod p$. In particular $N(\alpha)$ is a nonempty set.*

Lemma 4.6. *If A is conditioned upon a function $h : \{-1,1\}^t \to \{1,\zeta\}$ and $\hat{A}_\alpha \neq 0$, then for every $x \in N(\alpha)$, $h(x)$ is true, i.e., $h(x) = \zeta$.*

In this section our numbers are elements of the number field $\mathbb{Q}(\zeta)$, the rational numbers with the p^{th} root of unity added. We use the homomorphism σ_a, $0 \leq a \leq p-1$ which has the property that $\sigma_a(\zeta^i) = \zeta^{ia}$. For x a p^{th} root of unity we have $\sigma_a(x) = x^a$ but this is not true in general.

We have the following straightforward lemma of which we omit the proof.

Lemma 4.7. *For $x \neq 1$ a p^{th} root of unity we have*

$$\sum_{a=0}^{p-1} \sigma_a(x) = 0.$$

Finally define

$$\pi_p(\beta)(x) = \sum_{y \in \pi^{-1}(x)} \beta(y) \bmod p$$

as generalization of π_2. Lemma 2.8 generlizes.

Lemma 4.8. *Let $\beta \subseteq \{-1,1\}^W$, $U \subseteq W$ and $f : \{-1,1\}^U \to Z_p$, then*

$$\chi_\beta(f) = \chi_{\pi_p(\beta)}(f).$$

4.2 The basic test

We first define the basic test which is completely analogous to the binary case. We assume that tables A, B are folded over true and respect exponentiation. The table B (supposed long p-code on W) is conditioned upon the CNF formula φ_W.

Basic mod p PCP

1. V chooses U, W and φ_W as in 2PP(u).

2. V chooses two functions f and f' on U, taking values in Z_p uniformly at random.

3. V chooses two random functions g and g' on W taking values in Z_p uniformly at random. V defines a third function h by setting for each $y \in \{-1,1\}^W$, $h(y) = g(y) f(\pi(y))\ \text{mult}(g'(y), f'(\pi(y)))$.

4. V accepts iff $B(h) = B(g)A(f)\ \text{mult}(B(g'), A(f'))$.

Obviously the completeness of the basic test is 1 and we turn to the soundness.

Lemma 4.9. *If the verifier in the basic test accepts with probability $(1+\delta)/p$ then there exists a strategy for P_1 and P_2 in 2PP(u) that makes the verifier accept with probability $p^{-O(1)}\delta^{O(1)}$.*

First note that

$$B(h)^{-1}B(g)A(f)\ \mathrm{mult}(B(g'),A(f'))$$

is a p^{th} root of unity which is 1 iff the test accepts.

This implies, under the hypothesis of the lemma and using Lemma 4.7, that

$$E_{U,W,f,f',g,g'}[\sum_{a=1}^{p-1}\sigma_a(B(h)^{-1}B(g)A(f)\ \mathrm{mult}(B(g'),A(f')))] = \delta.$$

Using 4.1 and the fact that our tables respect exponentiation we see that

$$E_{U,W,f,f',g,g'}[\sum_{b=0}^{p-1}\sum_{c=0}^{p-1}\sum_{a=1}^{p-1}B(h)^{-a}B(g^a)A(f^a)B(g'^{ab})A(f'^{ac})\zeta^{-bc}] = p\delta.$$

We conclude that there must be some value of (a,b,c) such that

$$|E_{U,W,f,f',g,g'}[B(h)^{-a}B(g^a)A(f^a)B(g'^{ab}),A(f'^{ac})]| \geq p^{-2}\delta. \tag{4.1}$$

Replacing (h,g,f,g',f') by $(h^a,g^a,f^a,g'^a,f'^{)}$ preserves probability and hence changing the value of c, we can without loss of generality assume that $a=1$.

Fix U,W,f' and g' and let us study

$$E_{f,g}[B(h^{-1})B(g)A(f)]. \tag{4.2}$$

Replacing each function by its Fourier expansion we see that this equals

$$\sum_{\beta_1,\beta_2,\alpha}\hat{B}_{\beta_1}\hat{B}_{\beta_2}\hat{A}_{\alpha}E_{f,g}[\chi_{\beta_1}(f^{-1}g^{-1}\mathrm{mult}(f',g')^{-1})\chi_{\beta_2}(g)\chi_{\alpha}(f)].$$

The inner expected value is 0 unless $\beta_1=\beta_2$ and $\pi_p(\beta_1)=\alpha$ and hence (4.2) equals

$$\sum_{\beta}\hat{B}_{\beta}^2\hat{A}_{\pi_p(\beta)}\chi_{\beta}(\mathrm{mult}(f',g')^{-1}). \tag{4.3}$$

Returning to (4.1) we need to analyze

$$E_{f',g'}[\chi_{\beta}(\mathrm{mult}(f',g')^{-1})B(g'^{b})A(f'^{c})]. \tag{4.4}$$

Fix the value of f'. When $b=0$, averaging over g' gives 0 unless $f'(\pi(z))=1$ for all $z\in N(\beta)$. The probability of picking such an f' is $p^{-|\pi(N(\beta))|}$. Now consider the case when $b\neq 0$. Define β' as follows : for every y, $\beta'(y)=b^{-1}e(y)\beta(y)$ where $f'(\pi(y))=\zeta^{e(y)}$. Averaging (4.4) over g' yields $\hat{B}_{\beta'}A(f'^{c})$.

We note that $f's$ which are different on $\pi(N(\beta))$ give different β'. Let Δ_β be the set of all possible β'. We have $|\Delta_\beta|=p^{|\pi(N(\beta))|}$ and over all the choices of f', every $\beta'\in\Delta_\beta$ occurs equally often. Using this observation and applying Cauchy-Schwartz inequality gives

$$\begin{aligned}|\ E_{f'}[\hat{B}_{\beta'}A(f'^{c})]\ | \ \leq\ E_{f'}[|\hat{B}_{\beta'}|] \quad &= \quad p^{-|\pi(N(\beta))|}\sum_{\beta'\in\Delta_\beta}|\hat{B}_{\beta'}| \ \leq \\ p^{-|\pi(N(\beta))|/2}\left(\sum_{\beta'\in\Delta_\beta}|\hat{B}_{\beta'}|^2\right)^{1/2} \quad &\leq \quad p^{-|\pi(N(\beta))|/2}.\end{aligned}$$

This implies that we get an overall upper bound on the expectation of (4.1) as

$$E_{U,W}\left[\sum_{\beta} |\hat{B}_{\beta}|^2\ |\hat{A}_{\pi_p(\beta)}|\ p^{-|\pi(N(\beta))|/2}\right].$$

Now we can extract prover strategies in a similar way as in the proof of Lemma 3.2, making use of (3.7). A minor difference is that now α (β) are functions and not sets. The provers pick α (β) with probability $\hat{A}^2_{\alpha}$ ($\hat{B}^2_{\beta}$) and pick a random $x \in N(\alpha)$ (a random $y \in N(\beta)$).

4.2.1 Iterated tests

The basic test in the previous section can be iterated in a way similar to the Section 3.2. We have only attempted the simpler analysis of almost disjoint sets and this is what we present here.

4.2.2 The almost disjoint sets test

We first define the test which is an iteration of the basic test studied in the last section.

k-iterated mod p almost disjoint sets PCP

1. V chooses U as in 2PP(u).
2. V chooses independently k sets $(W_l)_{l=1}^{k}$, that can appear with U in 2PP(u). Each W_l is chosen with the distribution induced by 2PP(u), i.e., the distribution of the pair U, W_l is the same as the distribution of U, W in 2PP(u).
3. V chooses $2k$ functions $(f_i)_{i=1}^{k}$ and $(f'_j)_{j=1}^{k}$ on U taking values in Z_p uniformly at random.
4. For each l, $1 \le l \le k$, V chooses two functions g_l and g'_l on W_l taking values in Z_p uniformly at random.
5. For each triple i, j, l such that $i + j + l \equiv 0 \bmod k$ define a function h_{ijl} by setting for each $y \in \{-1,1\}^{W_l}$, $h_{ijl}(y) = g_l(y) f_i(\pi(y))\, \mathrm{mult}(g'_l(y), f'_j(\pi(y)))$.
6. V accepts iff $B_{W_l}(h_{ijl}) = B_{W_l}(g_l) A_U(f_i)\, \mathrm{mult}(B_{W_l}(g'_l), A_U(f'_j))$ for all $i + j + l \equiv 0 \bmod k$.

We have the following theorem.

Theorem 4.10. *The almost disjoint sets test in Z_p has completeness* 1 *and soundness $p^{-k^2} + p^{O(1)} d_c^{\Omega(u)}$, where d_c is the constant from Theorem 2.1.*

Proof. The completeness is obvious and we need to analyze the soundness. To this end let

$$\mathrm{Acc}(i,j,l) = B_l(h_{ijl})^{-1} B_l(g_l) A(f_i)\ \mathrm{mult}(A(f'_j), B_l(g'_l)),$$

which is 1 if the test associated with (i, j, l) accepts and otherwise it is a different p^{th} root of unity.

Let Z_0 be the set of all triples (i,j,l) with $i+j+l \equiv 0(\bmod\ \mathrm{k})$ and let $S \in GF(p)^{k^2}$ be a vector whose coordinates are indexed by the triples in Z_0. We have

$$\prod_{(i,j,l)\in Z_0} \frac{\sum_{a=0}^{p-1}(\mathrm{Acc}(i,j,l))^a}{p} = p^{-k^2} \sum_{S\in GF(p)^{k^2}} \prod_{(i,j,l)\in Z_0} (\mathrm{Acc}(i,j,l))^{S(i,j,l)}.$$

By Lemma 4.7 this expression equals 1 if the test accepts and is 0 otherwise and thus its expected value is the probability that the test accepts. The term with $S \equiv 0$ is 1 and to establish the theorem it is sufficient to establish that any term with $S \not\equiv 0$ is upper bounded above by $p^{O(1)} d_c^{\Omega(u)}$. Let T_S be the expected value of the term corresponding to S. We go on to establish a strategy for P_1 and P_2 which makes the verifier in $2\mathrm{PP}(u)$ accept with probability $p^{-O(1)}|T_S|^{O(1)}$.

Suppose without loss of generality that $S(k,k,k) = r \neq 0$ and fix the values of f_i, $i \neq k$, f_j, $j \neq k$ and (W_l, g_l, g'_l) for $l \neq k$ in such a way as not to decrease $|T_S|$. Since we only have one remaining function of each type let us for readability discard the index.

By Lemma 4.1 and from the fact that any other triple intersects with the given triple in at most one place we conclude that T_S, after the above fixings, can be written as the sum of p^2 terms of the form

$$B(h)^{-r} A'(f) A''(f') C(g,g'), \tag{4.5}$$

each with a coefficient of complex absolute value $1/p$. Here A', A'', B, and C takes values which are p^{th} roots of unity. We conclude that there is such an expression of the form (4.5) whose expectation over U, W, h, f, f', g, and g' is at least $|T_S|/p$.

Here A' and A'' are functions that only depend on U and hence might be used to extract strategy for P_2. B is the original long p-code on $W = W_k$ and hence is useful for extracting strategy for P_1.

We now want to compute the expected value of this expression over random choices of f, f', g and g'. Expanding all factors except $A''(f')$ by the Fourier transform we get

$$\sum_{\alpha,\beta,\gamma,\gamma'} \hat{A}'_\alpha \hat{B}_\beta \hat{C}_{\gamma,\gamma'} E[\chi_\alpha(f)\chi_{-r\beta}(gf\ \ \mathrm{mult}(f',g'))\chi_\gamma(g)\chi_{\gamma'}(g')A''(f')]. \tag{4.6}$$

Now taking the expected value over f we see that unless $\alpha = r\pi_p(\beta)$ the term is 0. Similarly we need $\gamma = r\beta$. Fix f' and define β' as follows : for every y, $\beta'(y) = re(y)\beta(y)$ where $f'(\pi(y)) = \zeta^{e(y)}$. With this definition, we have

$$\chi_{-r\beta}(\mathrm{mult}(f',g')) = \chi_{-\beta'}(g')$$

Thus unless $\gamma' = \beta'$, the expectation is 0. Thus (4.6) equals

$$\sum_\beta \hat{A'}_{r\pi_p(\beta)} \hat{B}_\beta \hat{C}_{\beta,\beta'} A''(f') \tag{4.7}$$

Note that β' is uniquely determined by β and f' and functions f' which are different on $\pi(N(\beta))$ give different $\beta's$. Let Δ_β be the set of all possible β's. We have $|\Delta_\beta| = p^{|\pi(N(\beta))|}$ and over all the choices of f', every $\beta' \in \Delta_\beta$ occurs equally often. This implies that

$$\begin{aligned} |\ E_{f'}[\hat{C}_{\beta,\beta'(\beta,f')} A''(f')]\ | \quad &\le \quad E_{f'}[|\hat{C}_{\beta,\beta'(\beta,f')}|] \ \le \\ p^{-|\pi(N(\beta))|} \sum_{\beta'\in\Delta_\beta} |\hat{C}_{\beta,\beta'}| \quad &\le \quad p^{-|\pi(N(\beta))|/2} (\sum_{\beta'\in\Delta_\beta} |\hat{C}_{\beta,\beta'}|^2)^{1/2}. \end{aligned} \tag{4.8}$$

Substituting this estimate into (4.7) and using Cauchy-Schwartz inequality over β we get the upper estimate

$$\left(\sum_{\beta} |\hat{B}_{\beta}|^2 |\hat{A}'_{r\pi_p(\beta)}|^2 p^{-|\pi(N(\beta))|}\right)^{1/2} \left(\sum_{\beta,\beta' \in \Delta_\beta} |\hat{C}_{\beta,\beta'}|^2\right)^{1/2} \leq \left(\sum_{\beta} |\hat{B}_{\beta}|^2 |\hat{A}'_{r\pi_p(\beta)}|^2 p^{-|\pi(N(\beta))|}\right)^{1/2}$$

for $|T_S|/p$. The same strategy as defined in the basic test now makes the verifier accept in $2\mathrm{PP}(u)$ with probability $p^{-O(1)}|T_S|^{O(1)}$ and the theorem follows. □

5 Conclusions

We have established that the query efficient test of Samorodnitsky and Trevisan can be extended to include perfect completeness in several different ways. The tests are simple and the analyses are only moderately complicated, in particular the proofs using the approach of [15] are fairly straightforward.

All this taken together gives us good hope that, in the not too distant future, we will see more powerful PCPs with even more applications to inapproximability of NP-hard optimization problems. In particular the fact that we can include perfect completeness gives hope that stronger lower bounds for coloring of graphs of small chromatic number could be possible. Clearly to obtain such results obstacles of other nature need also be overcome. We note that some progress for constant colorable graphs has already occurred [9], but getting strong results for 3-colorable graphs seems to require new ideas.

6 Acknowledgments

We would like to thank Sanjeev Arora for several helpful discussions. We are also grateful to the anonymous referees for many comments that helped to improve the quality of the paper.

References

[1] S. ARORA, C. LUND, R. MOTWANI, M. SUDAN, AND M. SZEGEDY: Proof verification and the hardness of approximation problems. *JACM: Journal of the ACM*, 45:501–555, 1998. 1, 2.2

[2] S. ARORA AND S. SAFRA: Probabilistic checking of proofs: A new characterization of NP. *JACM: Journal of the ACM*, 45:70–122, 1998. 1

[3] M. BELLARE, O. GOLDREICH, AND M. SUDAN: Free bits, PCPs, and nonapproximability–towards tight results. *SICOMP: SIAM Journal on Computing*, 27:804–915, 1998. 1, 2.3

[4] U. FEIGE: A threshold of ln n for approximating set cover. *JACM: Journal of the ACM*, 45:634–652, 1998. 1, 2.2

[5] U. FEIGE, S. GOLDWASSER, L. LOVASZ, S. SAFRA, AND M. SZEGEDY: Interactive proofs and the hardness of approximating cliques. *JACM: Journal of the ACM*, 43:268–292, 1996. 1

[6] V. GURUSWAMI, J. HÅSTAD, AND M. SUDAN: Hardness of approximate hypergraph coloring. *SICOMP: SIAM Journal on Computing*, 31:1663–1686, 2002. 1, 1

[7] V. GURUSWAMI, D. LEVIN, M. SUDAN, AND L. TREVISAN: A new characterization of np with 3 query pcps. In *Proceedings of 39th Annual IEEE Symposium of Foundations of Computer Science*, pp. 8–17, 1998. 1, 1, 3.2.3, 3.2.3

[8] J. HÅSTAD AND S. KHOT: Query efficient pcps with perfect completeness. In *In Proceedings of 42nd Annual IEEE Symposium of Foundations of Computer Science*, pp. 610–619, 2001. 1.1, 3.2.3

[9] S. KHOT: Improved inapproximability results for maxclique, chromatic number and approximate graph coloring. In *Proceedings of 42nd Annual IEEE Symposium of Foundations of Computer Science*, pp. 600–609, 2001. 1, 1.6, 5

[10] S. KHOT: Hardness results for coloring 3-colorable 3-uniform hypergraphs. In *Proceedings of 43rd Annual IEEE Symposium on Foundations of Computer Science*, pp. 23–32, 2002. 3.2.3

[11] R. RAZ: A parallel repetition theorem. *SIAM Journal on Computing*, 27:763–803, 1998. 2.2

[12] A. SAMORODNITSKY AND L. TREVISAN: A pcp characterization of np with optimal amortized query complexity. In *Proceedings of the 32nd Annual ACM Symposium on Theory of Computing*, pp. 191–199, 2000. (document), 1, 2.2, 3.2, 3.2.2, 3.5

[13] J. HÅSTAD: Clique is hard to approximate within $n^{1-\varepsilon}$. *Acta Mathematica*, 182:105–142, 1999. 1

[14] J. HÅSTAD: Some optimal inapproximability results. *Journal of the ACM*, 48:798–859, 2001. 1, 2.3.1, 3.1, 3.2.3

[15] J. HÅSTAD AND A. WIGDERSON: Simple analysis of graph tests for linearity and pcp. *Random Structures and algorithms*, 22:139–160, 2003. 1, 3.2, 5

[16] L. TREVISAN: Approximating satisfiable satisfiability problems. *Algorithmica*, 28:145–172, 2000. 1

AUTHORS[1]

Johan Håstad
Professor
Royal Instiute of Technology, Stockholm, Sweden
johanh [at] nada [dot] kth [dot] se
http://www.nada.kth.se/~johanh

[1] To reduce exposure to spammers, THEORY OF COMPUTING uses various self-explanatory codes to represent “AT” and “DOT” in email addresses.

Subhash Khot
Assistant Professor
Georgia Instiute of Technology, Atlanta GA-30332
khot [at] cc [dot] gatech [dot] edu
http://www.cc.gatech.edu/~khot

ABOUT THE AUTHORS

JOHAN HÅSTAD graduated from M.I.T. in 1986. His advisor was Shafi Goldwasser. His CS interests include cryptograpy, complexity theory and approximability of NP-hard optimization problems. He also enjoys table tennis and wine.

SUBHASH KHOT graduated from Princeton University in 2003 under the supervision of Prof. Sanjeev Arora. He is interested in complexity theory, approximability of NP-hard problems and theory of metric embeddings. He loves watching movies, cooking, and hanging out with friends and the family. Once an avid fan of cricket, he is no more interested in the game, thanks to the dismal performance of the Indian cricket team in last several years.