Electronic Colloquium on Computational Complexity, Report No. 94 (2001)

Vertex Cover on 4-regular Hyper-graphs is Hard to Approximate Within $2 - \epsilon$

Jonas Holmerin
Royal Institute of Technology
SE-100 44 Stockholm, SWEDEN
E-mail: joho@nada.kth.se
Phone: +46 8 790 68 09   Fax: +46 8 790 09 30

December 3, 2001

Abstract

We prove that Minimum Vertex Cover on 4-regular hyper-graphs (or in other words, Minimum Hitting Set where all sets have size exactly 4), is hard to approximate within $2-\epsilon$. We also prove that the maximization version, in which we are allowed to pick $B = pn$ elements in an n-vertex hyper-graph, and are asked to cover as many edges as possible, is hard to approximate within $1/(1-(1-p)^4) - \epsilon$ when $p \geq 1/2$ and within $((1-p)^4 + p^4)/(1-(1-p)^4) - \epsilon$ when $p < 1/2$. From this follows that the general problem when B is part of the input is hard to approximate within $16/15 - \epsilon$. These results also hold for k-regular hyper-graphs when $k > 4$.

1 Introduction

Consider Minimum Vertex Cover, i.e, the problem in which we are given a graph and are asked to find a minimum set of vertices S such that each edge has an end-point in S.

It is well-known that this problem is **NP**-hard to solve exactly, and it was proven by Håstad [9] that it is **NP**-hard to approximate within $7/6 - \epsilon$. This was recently improved to $10\sqrt{5} - 21 - \epsilon$ by Dinur and Safra [4]. The best known algorithm approximates the problem within a factor $2 - o(1)$

Now consider the generalization to k-regular hyper-graphs, or equivalently, Minimum Hitting Set where all sets has size exactly k. We call this problem Minimum Ek Hitting Set. Since the general Minimum Hitting Set problem is equivalent to Minimum Set Cover, it follows by a result of Feige [5] that Minimum Hitting Set is "almost" **NP**-hard to approximate within a factor $(1-\epsilon)\ln n$ for any $\epsilon > 0$. This result is essentially tight since there is an $1 + \ln n$-approximation algorithm [10].

A less well studied case is what happens when k is a constant not equal to 2 (i.e, Minimum Vertex Cover extended to k-uniform hyper-graphs). There is a trivial approximation preserving reduction from Minimum Ek Hitting Set to Minimum E$k+1$ Hitting Set (just add one unique element to each set), so

ISSN 1433-8092

the hardness result for Minimum Vertex Cover also applies to Minimum Ek Hitting Set for $k > 2$. The best known algorithm approximates Minimum Ek Hitting Set within $k - \frac{(k-1)\ln\ln n}{\ln n}$ [8], so this is even farther from tight than for Minimum Vertex Cover. Since the general problem where the size of the sets is unbounded is hard to approximate within a logarithmic factor, we would expect Minimum Ek Hitting Set to get harder to approximate as k grows. Indeed, recently Trevisan [14] proved that asymtoptically Minimum Ek Hitting Set is **NP**-hard to approximate within $\Omega(k^{1/19})$. For large values of k, this gives a stronger inapproximability result than what we get from Minimum Vertex Cover. In this paper we study the case $k = 4$, and we prove that this problem is **NP**-hard to approximate within $2 - \epsilon$. As mentioned above this bound also applies to $k > 4$, but for large constants the bound of [14] dominates.

We also consider the maximization version in which the input is extended with a number B and we are asked to find a subset of size B which intersects as many S_i as possible. A tight bound for the approximability of the general problem is $e/(e-1) - \epsilon$ [5], and it is known that the version in which all sets has size 2 is **APX**-hard [12]. Again we consider the version where all sets has size exactly 4 and we prove an explicit bound of $16/15 - \epsilon$ for the approximability of this problem.

The tool which we use is Probabilistically Checkable Proofs (PCPs). In the PCP model, membership in a language is checked by a probabilistic verifier which is given oracle access to the proof and which is allowed to accept incorrect proofs with some probability. The connection between PCPs and approximability was discovered by Feige et al [6] who proved that all languages in **NP** can be checked by a verifier which uses very few random bits and queries very few bits from the proof. They then used this result to prove inapproximability for Maximum Clique. Strengthening this result, the PCP theorem of Arora and Safra and Arora et al [2, 1] says that it suffices if the verifier uses $O(\log n)$ random bits and queries $O(1)$ bits from the proof. The above mentioned papers derive inapproximability results from PCPs by essentially using the PCP as a black box. The idea is that we automatically get a hard approximation problem from a PCP; namely the problem of computing a proof with maximum acceptance probability. This problem can then be reduced to the problem at hand. By specially designing the PCPs to make this reduction more efficient for the specific problem, Bellare et al [3] were able to get improved results for many problems, and Håstad [9] improved previous techniques to get almost optimal bounds for several problem.

In this paper we take a somewhat different approach. For some problems the natural optimization problem associated with a PCP is not computing the proof with maximum acceptance probability. For example, when studying graph coloring, something called the *covering complexity* turns out to be the right thing to study [7]. When considering Minimum E4 Hitting Set, we are interested in the optimization problem of computing the proof which is accepted with probability 1 and which has a minimal number of 1-bits. In the proof systems mentioned above *no* proof of an incorrect statement is accepted with probability 1 so this for them this makes no sense, but our proof system will have the property that the constant 1 proof is always accepted. Thus we are interested in the bounding acceptance probability when the number of 1-bits in the proof is bounded. A technical contribution of this paper is a tight analysis of this probability as function of the proportion of 1-bits in the proof.

2 Preliminaries

2.1 Optimization and Approximation

In this paper, we study polynomial time approximation algorithms for some **NP**-hard optimization problems. To measure the efficiency of such an algorithm, we prove guarantees of the form that the algorithm always outputs a feasible solution with weight at most some factor from the weight of the optimal solution.

Definition 1. Let P be a maximization problem. For an instance x of P let $\operatorname{opt}(x)$ be the optimal value. A solution y, with weight $w(x,y)$, is *c-approximate* if it is feasible and $w(x,y) \geq \operatorname{opt}(x)/c$.

Definition 2. Let P be a minimization problem. For an instance x of P let $\operatorname{opt}(x)$ be the optimal value. A solution y, with weight $w(x,y)$, is *c-approximate* if it is feasible and $w(x,y) \leq c \cdot \operatorname{opt}(x)$.

Definition 3. A *c-approximation algorithm* for an optimization problem is a polynomial time algorithm that for any instance x of the problem and any input y outputs a c-approximate solution.

We use the wording *to approximate within c* as a synonym for *to compute a c-approximate solution.*

We now turn to the optimization problems we study in this paper.

Definition 4. Minimum Hitting Set is the following minimization problem: Given a set X and m sets $S_1, \ldots, S_m$, where $S_i \subset X$, find a set $S \subset X$ of such that $S_i \cap S \neq \emptyset$ for all S_i and $|S|$ is minimized.

Definition 5. Minimum Ek Hitting Set is the following minimization problem: Given a set X and m sets $S_1, \ldots, S_m$, where $S_i \subset X$ and $|S_i| = k$, find a set $S \subset X$ such that $S_i \cap S \neq \emptyset$ for all S_i and $|S|$ is minimized.

Definition 6. Maximum Ek Set Hitting is the following maximization problem: Given a set X and m sets $S_1, \ldots, S_m$, where $S_i \subset X$ and $|S_i| = k$ and a positive integer B, find a set $S \subset X$ such that $|S| = B$ such that $|\{i | S_i \cap S \neq \emptyset\}|$ is maximized.

Definition 7. G-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 G of the clauses in ϕ are satisfiable and are supposed to decide if the formula is satisfiable.

2.2 Probabilistic Proof Systems and Approximability

A language L is in the class **NP** if there exists a polynomial time Turing machine M, with the properties that

1. For $x \in L$, there exists a proof π, of size polynomial in $|x|$, such that M accepts (x, π).

2. For $x \notin L$, M does not accept (x, π) for any proof π of size polynomial in $|x|$.

Arora and Safra [2] used a generalization of the above definition of **NP** to define the class $\mathbf{PCP}[r, q]$ (which was implicit in previous work), consisting of languages which have membership proofs that can be checked by a probabilistically verifier which has oracle access to the membership proof, is allowed to use r random bits, and allowed to query q bits from the oracle.

Definition 8. A probabilistic polynomial time Turing machine V with oracle access to π is an *(r, q)-restricted verifier* if it, for every oracle π and every input of size n, uses at most $r(n)$ random bits and queries at most $q(n)$ bits from the oracle. We denote by V^{π} the verifier V with the oracle π fixed.

Definition 9. A language L belongs to the class $\mathbf{PCP}_{c,s}[r, q]$ if there exists an (r, q)-restricted verifier V with the properties that

1. For $x \in L$, $\Pr_{\rho}[V^{\pi} \text{ accepts } (x, \rho)] \geq c$ for some oracle π.
2. For $x \notin L$, $\Pr_{\rho}[V^{\pi} \text{ accepts } (x, \rho)] \leq s$ for all oracles π.

where ρ is the random string of length r.

We call c and s the completeness and the soundness of the verifier, respectively. When $c = 1$, we say that the verifier has *perfect completeness.*

As a shorthand, we write $\mathbf{PCP}[q, r]$ for $\mathbf{PCP}_{1,1/2}[q, r]$.

As mentioned in the introduction, the connection between PCPs and approximability was first discovered by Feige et al. [6], who showed that

$$\mathbf{NP} \subseteq \mathbf{PCP}[O(\log n \log\log n), O(\log n \log\log n)]$$

and used this characterization of **NP** and a reduction to show that unless **NP** admits algorithms which run in slightly super-polynomial time, Maximum Clique cannot be approximated within any constant in polynomial time.

Results giving stronger characterizations of **NP** in terms of $\mathbf{PCP}[r, q]$ were proven by Arora and Safra [2] and Arora et al. [1]. The following result is by [1]:

Theorem 2.1 (The PCP theorem).

$$\mathbf{NP} = \mathbf{PCP}[O(\log n), O(1)].$$

In other words, this remarkable theorem says that membership for **NP**-languages can be probabilistically checked by a verifier which uses logarithmic randomness, always accepts a correct proof, rejects incorrect proof with probability at least 1/2, and looks only at a constant number of bits of the proof.

There is an approximation preserving reduction from general E3-Sat formulas to formulas where each variable occurs in exactly five clauses [11]. Together with this reduction, the PCP theorem implies the following theorem:

Theorem 2.2. *There is a constant G such that G-gap E3-Sat-5 is* **NP**-*hard.*

From this result it follows that there is some constant G such that it is **NP**-hard to approximate 3-Sat within G.

2.3 The Long Code and Discrete Fourier Transforms

In the rest of the paper we will consider $\mathbb{Z}_2$ to consist of the elements $\{1, -1\}$. Addition thus becomes multiplication. Hence 1 takes the place of 0 and -1 takes the place of 1 in the definitions in Section 2.1.

Definition 10. If U is some set of variables taking values in $\{-1, 1\}$, we denote by $\{-1, 1\}^U$ the set of all possible assignments to those variables. Define $\mathcal{F}_U = \{f \colon \{-1, 1\}^U \to \{-1, 1\}\}$.

Consider a set W of variables and a subset U of W. Then for any assignment y to the variables in W, write $y|_U$ for the restriction of y to U. We define projection on subsets of assignments:

Definition 11. Let W be a set of variables, let U be a subset of W, and let $\beta \subseteq \{-1, 1\}^W$. define the *projection of β on U*

$$\pi^U(\beta) = \{x | \beta \text{ contains an } y \text{ such that } y|_U = x\}$$

We will use the long code, invented by Bellare et al [3]. We get the long code of an assignment by simply writing down the value of all possible boolean functions on the assignment. More formally,

Definition 12. The *long code* of an assignment $x \in \{-1, 1\}^U$ is a mapping $A_x \colon \mathcal{F}_U \to \{-1, 1\}$ where $A_x(f) = f(x)$.

To use the analysis methods of [9], we also need the Fourier inversion theorem on a function $A \colon \mathcal{F}_U \to \{-1, 1\}$. The idea is that we view A as an element of the vector space of functions from $\mathcal{F}_U$ to $\mathbb{C}$ and expand A in an orthogonal basis for that vector space. Let

$$\chi_\alpha(f) = \prod_{x \in \alpha} f(x)$$

$$\hat{A}_\alpha = 2^{-2^{|U|}} \sum_{f \in \mathcal{F}_U} A(f)\chi_\alpha(f).$$

Then we have:

Theorem 2.3 (Fourier inversion). *A function $A \colon \mathcal{F}_U \to \{-1, 1\}$ can be written as*

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

This theorem is an extremely useful tool since the basis functions $\chi_\alpha(f)$ are just products of long codes for different assignments to the variables in U. Thus a coefficient $\hat{A}_\alpha$ is the correlation of A with a certain product of long codes.

We also need Parseval's equality:

Theorem 2.4 (Parseval's equality). *Let $A \colon \mathcal{F}_U \to \{-1, 1\}$. Then*

$$\sum_{\alpha \subseteq \{-1,1\}^U} \hat{A}_\alpha^2 = 1.$$

Typically we will give a verifier oracle access to a table A which is supposed to be the long code of some string. It is sometimes useful to access A in certain ways to ensure that A satisfy certain properties.

Sometimes we know that A is supposed to the the long code of an assignment x satisfying that some function h, $h(x) = -1$ (e.g., x is a satisfying assignment to some CNF-formula). Then we will use an invention of Håstad [9], *called conditioning upon h*. First we will need to define pointwise logical and of functions. Let

$$(f \wedge h)(x) = \begin{cases} f(x) & \text{if } h(x) = -1 \\ 1 & \text{if } h(x) = 1 \end{cases}$$

Definition 13. Given a function $A\colon \mathcal{F}_U \to \{-1, 1\}$ we define A_h, A *conditioned upon* h, by for each pair of functions $A_h(f) = A(f \wedge h)$.

We have the following lemma from [9]

Lemma 2.5. *Let $B = A_h$, where $A\colon \mathcal{F}_U \to \{-1, 1\}$. Then for any α such that there is $x \in \alpha$ with $h(x) = 1$, $\hat{B}_\alpha = 0$.*

This is an extremely useful property, since it means that A_h is not at all correlated with products of long codes of assignments where any of the assignments do not satisfy h.

3 PCP for Hitting Set

We construct a specially tailored PCP. The (now) standard approach is to construct a PCP such that the optimization problem of computing the proof which has maximum acceptance probability is easy to reduce to the problem at hand. In our case, we instead look at the optimization problem of the minimum number of -1 in the proof which makes the verifier accept with probability 1, and we design our PCP so that this problem is almost trivial to reduce to Minimum E4 Hitting Set. This means we want the test to be that not all bits are 1, and we want the verifier to have perfect completeness. We consider elements to be the positions in the proofs and we make a set for each random string consisting of the positions in the proof queried for that random string. Thus a correct proof will correspond to a hitting set of a size which are the number of occurrences of -1 in the proof. We have the following definition:

Definition 14. For $-1 \leq \rho \leq 1$, a proof is *ρ-balanced* if a fraction $\frac{1+\rho}{2}$ of the bits are 1. We say that a proof is *balanced* if it is 0-balanced.

Our encoding of the proofs will be such that a correct proof is balanced, and thus corresponds to a hitting set of $1/2$ of all elements. We want the verifier to have the property that if there is no correct proof, then no ρ-balanced proof makes the verifier accept with probability 1, for ρ as close to -1 as possible. This implies that when there is no correct proof, there is no hitting set of size a fraction $\frac{1-\rho}{2}$ of all elements. Note that the perfect completeness is crucial. The following lemma formalizes the above discussion.

Lemma 3.1. *Let $c > 0$ and suppose there is a $(O(\log n), O(1))$-restricted verifier for some **NP**-complete language L with the following properties:*

1. *The verifier V non-adaptively reads exactly k bits and accepts if not all are* 1.

2. *If the input is in L, then there is a* 0*-balanced proof which V accepts with probability* 1.

3. *If the input is not in L, then for $\rho \geq -c$, there is no ρ-balanced proof which V accepts with probability* 1.

Then Minimum Ek Hitting Set is impossible to approximate within $1 + c$ in polynomial time, unless $\mathbf{P} = \mathbf{NP}$.

Proof. On input x, create the following instance of Minimum Ek Hitting Set : For each bit $\pi(i)$ in the proof we have an element $x_i \in X$. For each random string of the verifier, we have a set S_i containing the elements corresponding to the bits the verifier reads. A proof π corresponds to a $S \subset X$ by $x_i \in S$ iff the corresponding bit $\pi(i)$ is -1. By the second property of V, if $x \in L$ is satisfiable, then there is a hitting set S of size less than or equal to $|X|/2$. By the third property of V, if $x \notin L$, no set of size smaller than $|X|(1 + c)/2$ is a hitting set. Thus it is **NP**-hard to approximate Minimum Ek Hitting Set within $1 + c$. □

The construction of the PCP generally proceeds as in [9]. By Theorem 2.2, any problem in **NP** can be reduced to G-gap E3-Sat-5. There is a well known one-round two-prover interactive proof system for G-gap E3-Sat-5 which we describe before proceeding.

3.1 An Interactive Proof System For G-gap E3-Sat-5

The two-prover one-round interactive proof system consists of two provers, P_1 and P_2, and one verifier. Given an instance, i.e., an E3-Sat formula ϕ, the verifier behaves as follows:

1. Pick a clause C and variable x in C uniformly at random from the instance.

2. Send x to P_1 and C to P_2. P_1 returns an assignment to x and P_2 returns an assignment to the variables in C.

3. Accept if these assignments are consistent and satisfy C.

When ϕ is satisfiable, the provers answers according to a satisfying assignment, and thus the verifier accepts with probability 1. Moreover, the provers can fool the verifier to accept an unsatisfiable instance of G-gap E3-Sat-5 with probability at most $(2 + G)/3$. To summarize this in the language of PCPs, the abovementioned proof system has completeness 1 and soundness $(2 + G)/3$. The soundness can be lowered to $((2+G)/3)^u$ by repeating the protocol u times independently in sequence, but it is also possible to construct a one-round proof system with low soundness 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 follows by a general result by Raz [13] that the soundness is at most c_G^u, where $c_G < 1$ is some constant depending on G but not on u or the size of the instance.

3.2 The PCP

The proof is a Standard Written Proof as defined in [9].

Definition 15. Let ϕ be a G-gap E3-Sat-5 formula with n variables $x_1, \ldots, x_n$, and $m = 5n/3$ clauses $C_1, \ldots, C_m$. Define $\mathcal{U}$ to be the set of all sets U of variables, where $|U| = u$. Similarly, define $\mathcal{W}$ to be the set of all sets W of clauses where $|W| = u$.

Definition 16. A *written proof with parameter* u contains for each set W of u disjoint clauses a string of length $2^{2^{3ku}}$ which is interpreted as the table of a function $A_W \colon \mathcal{F}_W \to \{-1, 1\}$.

Definition 17. A written 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_W is the long code $x|_W$.

Thus a correct proof is always balanced.

Lemma 3.2. *For a ρ-balanced written proof,*

$$\operatorname*{E}_W[\hat{A}_{W,\emptyset}] = \rho.$$

where the expectation is over sets W of u disjoint clauses.

Proof. We have that

$$\operatorname*{E}_W[\hat{A}_{W,\emptyset}] = \operatorname*{E}_{W,f}[A_W(f)] = \rho$$

This follows because since all tables have the same size, we have that $A_W(f)$ is an uniformly distributed bit in the proof. □

Our protocol will be almost identical to the one Håstad [9] uses to prove inapproximability for Maximum E4 Set Splitting. The only thing we change is the acceptance predicate. We will construct the verifier in two steps. The first verifier is given in Figure 1. The final verifier will use this one as a subroutine. The completeness of the test is straightforward:

Lemma 3.3. *If ϕ is satisfiable, there is a balanced proof which Test HS-ϵ accepts with probability* 1.

Turning to the soundness, first note that since the probability that the verifier accepts immediately in step 3 is $o(1)$, this case adds at most $o(1)$ to the soundness. Thus it suffices to analyze the verifier in the case when we have conditioned on that both W_1 and W_2 are sets of u disjoint clauses.

Below, all expectations over W_i are conditioned on that W_i contains u disjoint clauses, but in order to simplify the notation we omit this. Let $A_1 = A_{W_1}$

Test HS-ϵ. Input: A G-gap E3-Sat-5 formula $\phi = C_i \wedge \ldots \wedge C_m$ with n variables and m clauses and a Standard Written Proof with parameter u with each table A_W conditioned upon $\phi_W = \bigwedge_{C_i \in W} C_i$.

1. Select uniformly at random a set $U = \{x_1, \ldots, x_u\}$ of u variables.
2. Select two sets W_1 and W_2 in the following way: For each $x_j \in U$ pick uniformly at random a clause C_j in which x_j occurs. Let $W_1 = \{C_1, \ldots, C_u\}$. Repeat the process independently for W_2.
3. Accept if either W_1 or W_2 is not a set of u disjoint clauses.
4. Select uniformly at random $f \in \mathcal{F}_U$.
5. Select uniformly at random $g_i \in \mathcal{F}_{W_i}$.
6. Select $h_1 \in \mathcal{F}_{W_1}$, by for each y, if $f(y|_U) = -1$ set $h_1(y) = -g_1(y)$. If $f(y|_U) = 1$, set $h_1(y) = g_1(y)$ with probability $1 - \epsilon$ and $h_1(y) = -g_1(y)$ with probability ϵ.
7. Select $h_2 \in \mathcal{F}_{W_2}$, by for each y, if $f(y|_U) = 1$ set $h_2(y) = -g_2(y)$. If $f(y|_U) = -1$, set $h_2(y) = g_2(y)$ with probability $1 - \epsilon$ and $h_2(y) = -g_2(y)$ with probability ϵ.
8. Reject if
$$A_{W_1}(g_1) = A_{W_1}(h_1) = A_{W_2}(g_2) = A_{W_2}(h_2) = 1$$
else accept.

Figure 1: Verifier for Hitting Set – first step

and let $A_2 = A_{W_2}$. Note that the acceptance probability of the verifier can be written as

$$\mathrm{E}\left[1 - \frac{(1+A_1(g_1))(1+A_1(h_1))(1+A_2(g_2))(1+A_2(h_2))}{16}\right], \tag{1}$$

where the expectation is over $U, W_1, W_2, f, g_1, g_2, h_1$ and h_2. Now fix U and consider the terms in (1). Using Fourier expansion (Theorem 2.3), we have the following two lemmas (implicit in [9]):

Lemma 3.4.

$$\mathrm{E}[A_1(g_1)A_1(h_1)] = \mathrm{E}[A_2(g_2)A_2(h_2)] = \tag{2}$$

$$\mathop{\mathrm{E}}_{W_1}\left[\sum_{\beta} \hat{A}^2_{1,\beta} \prod_{x \in \pi^U(\beta)} (\frac{1}{2}((-1)^{s_x} + (1-2\epsilon)^{s_x}))\right]. \tag{3}$$

where s_x is the number of $y \in \beta$ such that $y|_U = x$, and the expectations is over $W_1, W_2, f, g_1, g_2, h_1$ and h_2.

Lemma 3.5.

$$\mathrm{E}[A_1(g_1)A_1(h_1)A_2(g_2)A_2(h_2)] = \tag{4}$$

$$\mathop{\mathrm{E}}_{W_1,W_2}\left[\sum_{\alpha,\beta} \hat{A}^2_{1,\alpha}\hat{A}^2_{2,\beta} \prod_{x} (\frac{1}{2}((-1)^{s_x}(1-2\epsilon)^{t_x} + (1-2\epsilon)^{s_x}(-1)^{t_x}))\right]. \tag{5}$$

Where s_x is the number of $y \in \alpha$ such that $y|_U = x$, t_x is the number of $y \in \beta$ such that $y|_U = x$ and the expectation is over $W_1, W_2, f, g_1, g_2, h_1$ and h_2.

Let

$$\nu(s) = \frac{1}{2}((-1)^s + (1-2\epsilon)^s)$$

and let

$$\kappa(s,t) = \frac{1}{2}((-1)^s(1-2\epsilon)^t + (1-2\epsilon)^s(-1)^t)$$

Let $0 < \delta < 1/2$ be a constant, and

$$T_U = \mathop{\mathrm{E}}_{W_1}\left[\sum_{\substack{|\beta| \le \delta\epsilon^{-1} \\ \text{all } s_x \text{ even}}} \hat{A}^2_{1,\beta} \prod_{x \in \pi^U(\beta)} \nu(s_x)\right].$$

Let $R_U = \mathrm{E}[A_1(g_1)A_1(h_1)] - T_U$. Similarly, let

$$F_U = \mathop{\mathrm{E}}_{W_1,W_2}\left[\sum_{\substack{|\alpha|,|\beta| \le \delta\epsilon^{-1} \\ \text{all } s_x \text{ and } t_x \text{ even}}} \hat{A}^2_{1,\alpha}\hat{A}^2_{2,\beta} \prod_{x} \kappa(s_x, t_x)\right]$$

and let $Q_U = \mathrm{E}[A_1(g_1)A_1(h_1)A_2(g_2)A_2(h_2)] - F_U$.

Lemma 3.6. *For any ρ, if the proof is ρ-balanced, then the acceptance probability of Test HS-ϵ is at most*

$$1 - \frac{(1+\rho)^4 - 6\,\mathrm{E}_U[|R_U|] + E_U[Q_U]}{16} + \frac{1}{16}\mathrm{E}\left[\sum_{\substack{\alpha\cap\beta\neq\emptyset \\ |\alpha|,|\beta|\leq\delta\epsilon^{-1}}} \hat{A}^2_{1,\alpha}\hat{A}^2_{2,\beta}\right]$$

$$+o(1)$$

Proof. Consider

$$\mathrm{E}[(1+A_1(g_1))(1+A_1(h_1))(1+A_2(g_2))(1+A_2(h_2))] \tag{6}$$

where the expectation is over $U, W_1, W_2, f, g_1, g_2, h_1, h_2$. We compare (6) with $(1+\rho)^4$. Fix U. Let $\rho_U = \mathrm{E}_{W_1,g_1}[A_1(g_1)]$. Then $\mathrm{E}_U[\rho_U] = \rho$ We have that the terms on the form $\mathrm{E}_{W_i}[A_{W_i}(g_i)]$ and $\mathrm{E}_{W_i}[A_{W_i}(h_i)]$ contributes $4\rho_U$. Since g_1 and g_2 are uniformly and independently chosen, we have that h_1 and h_2 are also uniformly and independently chosen. Since we also have that W_1 and W_2 are independent and have the same distribution when U is fixed, we have that

$$\mathrm{E}[A_1(h_1)A_2(h_2)] = \mathrm{E}[A_1(g_1)A_2(g_2)] = \mathrm{E}[A_i(g_i)A_j(h_j)] = \rho_U^2$$

where $i \neq j$. And we have that these terms contribute $4\rho_U^2$. Next consider the two terms on the form $\mathrm{E}[A_i(g_i)A_i(h_i)]$. Let

$$q_U = \mathrm{E}[A_1(g_1)A_1(h_1)] = \mathrm{E}[A_2(g_2)A_2(h_2)].$$

Thus the terms contribute $2q_U$ to the sum. Next consider the terms on the form $\mathrm{E}[A_j(g_j)A_i(g_i)A_i(h_i)]$. These contribute $4\rho_U q_U$ to the sum. By Lemma 3.4 we have that

$$q_U = \mathop{\mathrm{E}}_{W_1}\left[\sum_{\beta}\hat{A}^2_{1,\beta}\prod_{x\in\pi^U(\beta)}(\frac{1}{2}((-1)^{s_x} + (1-2\epsilon)^{s_x}))\right].$$

Then $R_U = q_U - T_U$ and $2q_U + 4\rho_U q_U \geq T_U + 4\rho_U T_U - 6|R_U|$. Note that T_U is positive and that since $\mathrm{E}_{W_1}[\hat{A}_{W_1,\emptyset}] = \rho_U$, and $\mathrm{E}[X^2] \geq \mathrm{E}[X]^2$, we have that

$$T_U \geq \mathop{\mathrm{E}}_{W_1}[\hat{A}^2_{1,\emptyset}] \geq \rho_U^2$$

Finally we have the term

$$\mathrm{E}[A_1(g_1)A_1(h_1)A_2(g_2)A_2(h_2)],$$

which is $F_U + Q_U$. Since when U is fixed, W_1 and W_2 are independently and uniformly chosen with the same distribution,

$$T_U^2 = \mathop{\mathrm{E}}_{W_1,W_2}\left[\sum_{\substack{|\alpha|,|\beta|\leq\delta\epsilon^{-1} \\ \text{all } s_x \text{ and } t_x \text{ even}}} \hat{A}^2_{1,\alpha}\hat{A}^2_{2,\beta}\prod_{x\in\pi^U(\beta)}\nu(s_x)\nu(t_x)\right]$$

When $\beta \cap \alpha = \emptyset$, then the corresponding terms in T_U^2 and F_U are equal. Thus

$$F_U \geq T_U^2 - \operatorname*{E}_{W_1,W_2}\left[\sum_{\substack{\pi^U(\alpha)\cap\pi^U(\beta)\neq\emptyset \\ |\alpha|,|\beta|\leq\delta\epsilon^{-1}}} \hat{A}_{1,\alpha}^2 \hat{A}_{2,\beta}^2\right]$$

To summarize the progress so far, we have that (6) is at least

$$\begin{aligned}&1 + 4\rho_U + 4\rho_U^2 + 2T_U + 4\rho T_U + T_U^2 \\ &-6|R_U| - Q_U - \mathrm{E}\left[\sum_{\substack{\pi^U(\alpha)\cap\pi^U(\beta)\neq\emptyset \\ |\alpha|,||\beta|\leq\delta\epsilon^{-1}}} \hat{A}_{1,\alpha}^2 \hat{A}_{2,\beta}^2\right] - o(1)\end{aligned} \tag{7}$$

When $\rho_U \geq 0$ it is easily seen that (7) is at least $(1+\rho_U)^4$, since $T_U \geq \rho_U^2$. For the case $\rho_U < 0$, consider $2x + 4\rho_U x + x^2$. As a function of x, this is increasing for $x \geq \rho_U^2$, and thus $2T_U + 4\rho T_U + T_U^2 \geq 2\rho_U^2 + 4\rho_U^3 + \rho_U^4$, and we have proven that when U is fixed (6) is at most

$$1 - \frac{(1+\rho_U)^4 - 6|R_U| - Q_U}{16} + \frac{1}{16}E_{W_1}\left[\sum_{\substack{\pi^U(\alpha)\cap\pi^U(\beta)\neq\emptyset \\ |\alpha|,|\beta|\leq\delta\epsilon^{-1}}} \hat{A}_{1,\alpha}^2 \hat{A}_{2,\beta}^2\right] + o(1)$$

Taking the expectation over U and noting that $\mathrm{E}[X^4] \geq \mathrm{E}[X]^4$, we are done. □

The rest of the analysis is very similar to the analysis of the protocol used to prove hardness for set splitting in [9]. Before we can proceed to analyze the rest of the terms we need a technical lemma from [9]. Fix W_1 and β and consider

$$\prod_{x\in\pi^U(\beta)} \frac{1}{2}((-1)^{s_x} + (1-2\epsilon)^{s_x})$$

We want to prove that for large β, this quantity is large with small probability over U. To this end let

$$S_\epsilon^U(\beta) = \epsilon \sum_x \min(s_x, \epsilon^{-1}).$$

This can be seen as a generalization of $|\pi^U(\beta)|$. We have from the proof of Lemma 6.8 in [9] that

$$\begin{aligned}&\sum \hat{A}_{1,\beta}^2 \left|\prod_{x\in\pi^U(\beta)} \frac{1}{2}((-1)^{s_x} + (1-2\epsilon)^{s_x})\right| \leq \\ &\quad\sum \hat{A}_{1,\beta}^2 e^{-S_\epsilon^U(\beta)/2}\end{aligned} \tag{8}$$

We want to prove that when $|\beta|$ is large, then $S_\epsilon^U(\beta)$ is unlikely to be small. This follows from a technical lemma of [9]. We will use a corollary, which is Corollary 6.10 in [9]:

Lemma 3.7. *Let W be a set of u disjoint clauses and let $\alpha \subseteq \{-1,1\}^W$. There is a constant c such that if for $a, b > 1$, we have $|\alpha| = (ab)^{1/c}\epsilon^{-1}$. Then*

$$\Pr[S_\epsilon^U(\alpha) \le b] \le a^{-1}.$$

A possible value for $c = 1/35$.

The following Lemma bounds the terms in R_U where $|\beta| \le \delta\epsilon^{-1}$ and some s_x is odd.

Lemma 3.8. *For any U and W_1,*

$$\left| \sum_{\substack{|\beta| \le \delta\epsilon^{-1} \\ \textit{some } s_x \textit{ odd}}} \hat{A}^2_{W_1,\beta} \prod_{x \in \pi^U(\beta)} \nu(s_x) \right| \le \delta$$

Proof. Since some s_x is odd, we have for this s_x, $s_x\epsilon \le (-1)^{s_x} + (1-2\epsilon)^{s_x} \le 0$, and thus the corresponding factor has absolute value at most $s_x\epsilon \le \delta$. Since the absolute values of the other factors are bounded by 1 and $\sum_\beta \hat{A}^2_{1,\beta} \le 1$, the lemma follows. □

The next lemma helps bound the terms in $\mathrm{E}_U[R_U]$ where $|\beta| > \delta\epsilon^{-1}$.

Lemma 3.9. *There is a constant c such that for any W_1,*

$$\mathop{\mathrm{E}}_U \left[\left| \sum_{|\beta| > \delta\epsilon^{-1}} \hat{A}^2_{W_1,\beta} \prod_{x \in \pi^U(\beta)} \nu(s_x) \right| \right] \tag{9}$$

is at most

$$2\delta + \sum_{\substack{\beta \\ \delta\epsilon^{-1} \le |\beta| \le (\delta^{-2})^{1/c}\epsilon^{-1}}} \hat{A}^2_{1,\beta} \tag{10}$$

Proof. We have to bound the terms where $|\beta| \ge (2\delta^{-2})^{1/c}\epsilon^{-1}$. We have from (8) that

$$\mathop{\mathrm{E}}_U \left[\sum \hat{A}^2_{1,\beta} \left| \prod_{x \in \pi^U(\beta)} \nu(s_x) \right| \right] \le \sum \hat{A}^2_{1,\beta} \mathop{\mathrm{E}}_U [e^{-S_\epsilon^U(\beta)/2}]. \tag{11}$$

By Lemma 3.7, with $a = \delta^{-1}$ and $b \ge 2\delta^{-1}$, the probability that $S_\epsilon^U(\beta) \le 2\delta^{-1}$ is at most δ. Thus the term corresponding to β in (11) is at most $(\delta + e^{-\delta^{-1}})\hat{A}^2_{1,\beta} \le 2\delta\hat{A}^2_{1,\beta}$, (where we used that $e^{-x} \le x^{-1}$ for $x \ge 0$), and hence the sum (11) is at most 2δ. □

Lemma 3.10. *There is a strategy for the two-prover one round protocol with success probability at least*

$$\mathop{\mathrm{E}}_{U,W_1,W_2} \left[\delta^{-2}\epsilon^2 \sum_{\substack{\pi^U(\alpha) \cap \pi^U(\beta) \ne \emptyset \\ |\alpha|,|\beta| \le \delta\epsilon^{-1}}} \hat{A}^2_{W_1,\alpha}\hat{A}^2_{W_2,\beta} \right]$$

Proof. On receiving U, the prover P_1 picks an W_1, and then α of size at most $\delta\epsilon^{-1}$ with probability proportional to $\hat{A}^2_{W_1,\alpha}$. Then P_1 picks an $y_1 \in \alpha$ and returns $x = y_1|_U$. The prover P_2, on receiving W selects a β with probability proportional to $\hat{A}^2_{W,\beta}$ and returns a random $y_2 \in \beta$. Then the probability of picking α and β with non-empty intersection is at least

$$\sum_{\pi^U(\alpha)\cap\pi^U(\beta)\neq\emptyset|\alpha|,|\beta|\leq\delta\epsilon^{-1}} \hat{A}^2_{W_1,\alpha}\hat{A}^2_{W_2,\beta},$$

and the probability of the provers picking the same element in the intersection is at least $\delta^{-2}\epsilon^2$. The lemma follows. □

The following lemma bounds the terms in $E_U[Q_U]$ where both α and β is of size at most $\delta\epsilon^{-1}$ and at least one s_x or t_x is odd.

Lemma 3.11. *Let Acc be the acceptance probability of the modified two-prover protocol. Then*

$$\operatorname*{E}_{U,W_1,W_2}\left[\sum_{\substack{\alpha,\beta\\ |\alpha|,|\beta|\leq\delta\epsilon^{-1}\\ \textit{some } s_x \textit{ or } t_x \textit{ odd}}} \hat{A}^2_{1,\alpha}\hat{A}^2_{2,\beta}\prod_x \kappa(s_x,t_x)\right]$$

is at least

$$-\delta-\delta^2\epsilon^{-2}Acc$$

Proof. We analyze

$$\prod_x \kappa(s_x,t_x) = \prod_x\left(\frac{1}{2}((-1)^{s_x}(1-2\epsilon)^{t_x} + (1-2\epsilon)^{s_x}(-1)^{t_x})\right) \tag{12}$$

First consider the case when $\pi^U(\beta)\cap\pi^U(\alpha)=\emptyset$. Then for each x at least one of s_x or t_x is 0. We may assume that for some x, s_x is odd and t_x is 0. Then the corresponding factor in (12) is $\frac{1}{2}(-1+(1-2\epsilon)^{s_x})$ which is at least $-\epsilon s_x$. Since we have that $-\delta \leq -\epsilon s_x \leq 0$, and the other factors in the product (12) have absolute value at most 1 and $\sum \hat{A}^2_{1,\alpha}\hat{A}^2_{2,\beta} \leq 1$, we have that

$$\begin{aligned}\sum_{\alpha,\beta}\hat{A}^2_{1,\alpha}\hat{A}^2_{2,\beta}\prod_x\kappa(s_x,t_x) \geq&\\ -\delta - \sum_{\pi^U(\alpha)\cap\pi^U(\beta)\neq\emptyset}\hat{A}^2_{1,\alpha}\hat{A}^2_{2,\beta}&\end{aligned} \tag{13}$$

when the sums are restricted to α and β of size at most $\delta\epsilon^{-1}$. Furthermore, it follows from Lemma 3.10 that

$$\delta^{-2}\epsilon^2 E_{U,W_1,W_2}\left[\sum_{\pi^U(\alpha)\cap\pi^U(\beta)\neq\emptyset}\hat{A}^2_{1,\alpha}\hat{A}^2_{2,\beta}\right] \leq Acc,$$

and thus (13) is at least $-\delta-\delta^2\epsilon^{-2}Acc$. □

The following lemma bounds the terms of $\mathrm{E}_U[Q_U]$ where at least one of α or β has size greater $\delta\epsilon^{-1}$.

Lemma 3.12. *There is a constant c such that*

$$\mathop{\mathrm{E}}_{U,W_1,W_2}\left[\sum_{\alpha,\beta}\hat{A}_{1,\alpha}^2\hat{A}_{2,\beta}^2\prod_x \kappa(s_x,t_x)\right] \geq$$

$$-4\delta - \mathop{\mathrm{E}}_{U,W_1,W_2}\left[2\left(\sum_{\substack{\beta \\ \delta\epsilon^{-1}<|\beta|\leq(2\delta^{-2})^{1/c}\epsilon^{-1}}}\hat{A}_{1,\beta}^2\right)\right]$$

where the sum on the left-hand side is over α and β where at least one of α and β has size greater than $\delta\epsilon^{-1}$.

Proof. Below, sums over α,β are over α and β which where at least one of α and β has size greater than $\delta\epsilon^{-1}$. First consider the terms where at least one of α and β is of size at least $(2\delta^{-2})^{1/c}\epsilon^{-1}$. Without loss of generality we can assume that

$$|\alpha| \geq (2\delta^{-2})^{1/c}\epsilon^{-1}.$$

Then

$$\left|\prod_x \kappa(s_x,t_x)\right| \leq e^{-S_\epsilon^U(\beta)}$$

and by reasoning as in Lemma 3.9 we have

$$\mathop{\mathrm{E}}_{U}\left[\sum_{\alpha,\beta}\hat{A}_{1,\alpha}^2\hat{A}_{2,\beta}^2\prod_x \kappa(s_x,t_x)\right] \geq -4\delta$$

when the sum is restricted to terms where at least one of α and β is of size at least $(2\delta^{-2})^{1/c}\epsilon^{-1}$.

Next consider the terms where at least one of α and β is in the interval $[\delta\epsilon^{-1}, (2\delta^{-2})^{1/c}\epsilon^{-1}]$. We have that the terms where $\delta\epsilon^{-1} \leq \beta \leq (2\delta^{-2})^{1/c}\epsilon^{-1}$ is at least

$$-\sum_{\substack{\beta \\ \delta\epsilon^{-1}<|\beta|\leq(2\delta^{-2})^{1/c}\epsilon^{-1}}}\hat{A}_{1,\beta}^2$$

and similarly for α. The lemma follows. □

The following lemma summarizes the work done so far:

Lemma 3.13. *Let Acc be the acceptance probability for the modified two-prover protocol. Then Test HS-ϵ accepts a ρ-balanced proof with probability at most*

$$1-\frac{(1+\rho)^4}{16}+\frac{23}{16}\delta+\frac{8}{16}\sum_{\substack{\beta \\ \delta\epsilon^{-1}<|\beta|\leq(2\delta^{-2})^{1/c}\epsilon^{-1}}}\hat{A}_{1,\beta}^2+\frac{3}{16}\delta^2\epsilon^{-2}Acc+o(1).$$

Test FHS-δ. Input: A G-gap E3-Sat-5 formula $\phi = C_i \wedge \ldots \wedge C_m$ with n variables and m clauses and a Written Proof with parameters k and u. with each table A_W conditioned upon $\phi_W = \bigwedge_{C_i \in W} C_i$.

1. Set $t = \lceil \delta^{-1} \rceil$, let $\epsilon_1 = \delta$ and
$$\epsilon_i = \delta^{1+2/c} 2^{-1/c} \epsilon_{i-1} \tag{16}$$
2. Choose $i \in \{1, \ldots t\}$ with uniform probability. Run test HS-ϵ_i.

Figure 2: Verifier for Hitting Set – the final step

Proof. Combining Lemma 3.8 and Lemma 3.9 we get:

$$\mathop{\mathrm{E}}_{U}[|R_U|] \leq 3\delta + \mathop{\mathrm{E}}_{U,W_1,W_2}\left[\sum_{\substack{\beta \\ \delta\epsilon^{-1} < |\beta| \leq (2\delta^{-2})^{1/c}\epsilon^{-1}}} \hat{A}_{1,\beta}^2\right]. \tag{14}$$

Similarly, Lemma 3.11 and Lemma 3.12 gets us:

$$\mathop{\mathrm{E}}_{U}[Q_U] \geq -5\delta - \delta^2\epsilon^{-2} Acc - 2 \mathop{\mathrm{E}}_{U,W_1,W_2}\left[\left(\sum_{\substack{\beta \\ \delta\epsilon^{-1} < |\beta| \leq (2\delta^{-2})^{1/c}\epsilon^{-1}}} \hat{A}_{1,\beta}^2\right)\right] \tag{15}$$

Inserting these bounds and the bound on $\mathrm{E}\left[\sum_{\substack{\pi^U(\alpha) \cap \pi^U(\beta) \neq \emptyset \\ |\alpha|, |\beta| \leq \delta\epsilon^{-1}}} \hat{A}_{1,\alpha}^2 \hat{A}_{2,\beta}^2\right]$ from Lemma 3.10 in the expression in Lemma 3.6, the lemma follows. □

Our final verifier is given in Figure 2.

Proposition 3.14. *If ϕ is satisfiable, then for $\rho \leq 0$ there is a ρ-balanced proof which Test FHS-δ accepts with probability* 1. *For $\rho > 0$, there is a proof which Test FHS-δ accepts with probability at least* $1 - \frac{(1+\rho)^4 - (1-\rho)^4}{16} - \delta$.

Proof. The case when $\rho \leq 0$ follows from Lemma 3.3. For the case when $\rho > 0$, we consider the proof where each table is as close to a correct proof as possible. That is, the prover begins with a satisfying assignment x to ϕ and A_W is made as close to $A_{x|_W}$ as possible by changing a fraction ρ randomly chosen -1-bits to get the correct balance. Note that in a correct proof with probability at least $\frac{1-\delta}{2}$ one bit is -1 in which case the probability that this bit has been changed is ρ and with probability at least $\frac{1-\delta}{2}$ three bits are -1 in which case the probability that all these bits have been changed are at most ρ^3. Thus the verifier accepts with probability at least

$$1 - \frac{8\rho + 8\rho^3 + 4\delta}{16} = 1 - \frac{(1+\rho)^4 - (1-\rho)^4}{16} - \delta.$$

□

Proposition 3.15. *For any $\delta_0 > 0$, there is a choice of the parameters δ, k and u such that, for large enough ϕ, if ϕ is not satisfiable, then Test FHS-δ accepts a ρ-balanced proof with probability at most*

$$1 - \frac{(1+\rho)^4}{16} + \delta_0$$

Proof. By Lemma 3.13 it follows that the verifier accepts with probability at most

$$\begin{aligned}&1 - \frac{(1+\rho)^4}{16} + \frac{23}{16}\delta + \\ &+\frac{8}{16}\frac{1}{t}\sum_{i=1}^{t} \sum_{\substack{\beta \\ \delta\epsilon_i^{-1} < |\beta| \le (2\delta^{-2})^{1/c}\epsilon_i^{-1}}} \hat{A}_{1,\beta}^2 + \\ &+\frac{3}{16}\frac{1}{t}\sum_{i=1}^{t} \delta^2\epsilon_i^{-2} Acc\end{aligned} \tag{17}$$

Since the intervals in (17) are disjoint,

$$\frac{1}{t}\sum_{i=1}^{t} \sum_{\substack{\beta \\ \delta\epsilon_i^{-1} < |\beta| \le (2\delta^{-2})^{1/c}\epsilon_i^{-1}}} \hat{A}_{1,\beta}^2 \le \frac{1}{t}\sum_{\beta}\hat{A}_{1,\beta}^2 = \frac{1}{t} \le \delta$$

So the probability that the verifier accepts is at most

$$1 - \frac{(1+\rho)^4}{16} + \frac{31}{16}\delta + \frac{3}{16}\delta^2\epsilon_t^{-2} Acc$$

Since $\delta^2\epsilon_t^{-2}$ is a constant which depends only on δ, and Acc is bounded by c_G^u when the formula is not satisfiable, we can choose u such that $3\delta^2\epsilon_t^{-2}Acc \le \delta$. Thus the acceptance probability is at most

$$1 - \frac{(1+\rho)^4}{16} + 2\delta.$$

Since δ was an arbitrary positive real constant, we are done. □

3.3 Hitting Set – Hardness of Approximation

We can now prove our main theorem:

Theorem 3.16. *For any $\epsilon > 0$, Minimum E4 Hitting Set is **NP**-hard to approximate within $2 - \epsilon$.*

Proof. By Proposition 3.14 and Proposition 3.15 for any $\epsilon > 0$ there is a choice of the parameters for the verifier FHS-δ such that, for $rho < 1 - \epsilon$, if ϕ is satisfiable, there is a balanced proof which is is accepted with probability, and if ϕ is not satisfiable, no ρ-balanced proof is accepted with probability 1. There is a technicality in that this holds for the proofs implicitly constructed by conditioning, and not for the original tables. However, note that, since the table A_W is conditioned upon ϕ_W, the only functions the verifier actually queries from the proof are functions on the form $g \wedge \phi_W$, Since the proof is conditioned upon ϕ_W and W is a set of disjoint clauses the conditioned proof retains the same balance when restricted to such functions. We can thus consider the proof presented to the verifier to only consist of the positions the verifier actually queries. Thus we have that the verifier FHS-δ fulfills the assumptions of Lemma 3.1 with $c = 1-\epsilon$, and we are done. □

For Maximum E4 Set Hitting, we have the following:

Theorem 3.17. *For any $\epsilon > 0$, Maximum E4 Set Hitting is **NP**-hard to approximate within $16/15 - \epsilon$.*

Proof. By Proposition 3.14 and Proposition 3.15 with $\rho = 0$ it is **NP**-hard to distinguish between the case that a there is a set of size $B = n/2$ which intersects all sets and the case that all sets of size $n/2$ intersects at most a fraction $15/16 + \delta$ of all sets. The theorem follows. □

The previous statement says that an algorithm which should work for any size B can at best approximate Maximum E4 Set Hitting within $16/15 + \epsilon$. However, we can also say something about the lower bound as a function of B:

Theorem 3.18. *Let $B = pn$ where p is some constant. For any $\epsilon > 0$ when $B \geq n/2$, it is **NP**-hard to approximate Maximum E4 Set Hitting within*

$$\frac{1}{1-(1-p)^4} + \epsilon.$$

*When $B < n/2$, Maximum E4 Set Hitting is **NP**-hard to approximate within*

$$\frac{1-(1-p)^4+p^4}{1-(1-p)^4} + \epsilon$$

Proof. Create the same instance of Maximum E4 Set Hitting as in Lemma 3.1. When $B \geq n/2$ we have from Proposition 3.14 that if ϕ is satisfiable then there is a set S of size B which intersects all sets. If ϕ is not satisfiable, then by Proposition 3.15 there is no set S of size B which intersects more than a fraction $1-(1-B/n)^4+\delta_0$ of the sets. Since δ_0 can be made arbitrarily small, this proves the case when $B \geq n/2$.

For the case when $B < n/2$, we have that if ϕ is satisfiable then there is a set S of size B which intersects a fraction $1-(1-p)^4+p^4+\delta$ of all sets. As before if ϕ is not satisfiable, there is no set S of size B which intersects more than a fraction $1-(1-B/n)^4+\delta_0$. From the proof of Proposition 3.15 we have $\delta < \delta_0$ and since δ_0 can be made arbitrarily small, the case when $B < n/2$ follows. □

4 Open Problems

There are several intriguing open problems regarding Minimum Ek Hitting Set. For instance, can we prove an $\Omega(k)$ lower bound on the approximability of Minimum Ek Hitting Set?

Can we find an approximation algorithm which does better than a factor k for some k? As far as the author is aware, all current approximation algorithm for Minimum Ek Hitting Set are straightforward generalizations of algorithms for Minimum Vertex Cover. It might be that we can do better if we concentrate on the case when k is a large constant.

Would, say, a lower bound of 4 for Minimum E4 Hitting Set imply a lower bound of 2 for Minimum Vertex Cover? More generally, is there an approximation preserving reduction from Minimum Ek Hitting Set to Minimum E$k-1$ Hitting Set?

Can we get a better lower bound than the one we get for Minimum Vertex Cover for $k = 3$?

5 Acknowledgments

I thank Johan Håstad for useful discussions.

References

[1] 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.

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

[3] Mihir Bellare, Oded Goldreich, and Madhu Sudan. Free bits, PCPs and non-approximability—towards tight results. *SIAM Journal on Computing*, 27(3):804–915, June 1998.

[4] Irit Dinur and Shmuel Safra. The importance of being biased. Manuscript, October 01.

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

[6] Uriel Feige, Shafi Goldwasser, László Lovász, Shmuel Safra, and Márió Szegedy. Interactive proofs and the hardness of approximating cliques. *Journal of the ACM*, 43(2):268–292, March 1996.

[7] Venkatesan Guruswami, Johan Håstad, and Mahdu Sudan. Hardness of approximate hypergraph coloring. In *41st Annual Symposium on Foundations of Computer Science*, pages 149–158, Redondo Beach, California, 12–14 November 2000. IEEE.

[8] Eran Halperin. Improved approximation algorithms for the vertex cover problem in graphs and hypergraphs. In *Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 329–337, San Francisco, California, 9–11 January 2000.

[9] 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.*

[10] David S. Johnson. Approximation algorithms for combinatorial problems. *Journal of Computer and System Sciences*, 9:256–278, December 1974.

[11] Christos H. Papadimitriou and Mihalis Yannakakis. Optimization, approximation, and complexity classes. *Journal of Computer and System Sciences*, 43(3):425–440, December 1991.

[12] Erez Petrank. The hardness of approximation: Gap location. *Computational Complexity*, 4(2):133–157, 1994.

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

[14] Luca Trevisan. Non-approximability results for optimization problems on bounded degree instances. In *Proceedings of the Thirty-third Annual ACM Symposium on Theory of Computing*, pages 453–461, Hersonissos, Crete, 6–8 July 2001.