Hardness Results for Coloring 3-Colorable 3-Uniform Hypergraphs *

Subhash Khot †
Department of Computer Science,
Princeton University
email : khot@cs.princeton.edu

Abstract

In this paper, we consider the problem of coloring a 3-colorable 3-uniform hypergraph. In the minimization version of this problem, given a 3-colorable 3-uniform hypergraph, one seeks an algorithm to color the hypergraph with as few colors as possible. We show that it is NP-hard to color a 3-colorable 3-uniform hypergraph with constantly many colors. In fact, we show a stronger result that it is NP-hard to distinguish whether a 3-uniform hypergraph with n vertices is 3-colorable or it contains no independent set of size δn for an arbitrarily small constant $\delta > 0$. In the maximization version of the problem, given a 3-uniform hypergraph, the goal is to color the vertices with 3 colors so as to maximize the number of non-monochromatic edges. We show that it is NP-hard to distinguish whether a 3-uniform hypergraph is 3-colorable or any coloring of the vertices with 3 colors has at most $\frac{8}{9} + \epsilon$ fraction of the edges non-monochromatic where $\epsilon > 0$ is an arbitrarily small constant. This result is tight since assigning a random color independently to every vertex makes $\frac{8}{9}$ fraction of the edges non-monochromatic.

These results are obtained via a new construction of a probabilistically checkable proof system (PCP) for NP. We develop a new construction of the PCP Outer Verifier. An important feature of this construction is smoothening of the projection maps.

Dinur, Regev and Smyth [6] independently showed that it is NP-hard to color a 2-colorable 3-uniform hypergraph with constantly many colors. In the "good case", the hypergraph they construct is 2-colorable and hence their result is stronger. In the "bad case" however, the hypergraph we construct has a stronger property, namely, it does not even contain an independent set of size δn.

*Full version of this paper is available at http://www.cs.princeton.edu/∼khot/publications.html

†This work was done when the author was visiting the Dept of Computer Science, University of California, Berkeley.

1. Introduction

In this paper, we consider the problem of coloring 3-colorable 3-uniform hypergraphs. We consider both the minimization and maximization versions.

A q-uniform hypergraph $G = (V, E)$ consists of a set of vertices V and a set of edges E. Every edge $e \in E$ is a size-q subset of the set of vertices (so graphs are 2-uniform hypergraphs). A hypergraph is said to be k-colorable if the vertices can be colored with k colors so that for every edge, not all of its vertices have the same color.

Minimization Version : In the minimization version of the problem (also called Approximate Coloring problem), we are given a k-colorable (hyper)graph where k is a small constant and we seek an algorithm to color the (hyper)graph with as few colors as possible. It is well-known that it is NP-hard to test 3-colorability of graphs whereas Lovász [19] showed that it is NP-hard to test 2-colorability of 3-uniform hypergraphs. The known algorithms color k-colorable (hyper)graphs with $n^{\Omega(1)}$ colors. Table 1 gives (only) the best known algorithmic results.

There is a huge gap between these algorithmic results and the corresponding known hardness results. For 3-colorable graphs, we only know that it is hard to color them with 4 colors (see [14]). A big open problem is whether 3-colorable graphs are hard to color with constantly many colors. This problem seems well-beyond the scope of the current techniques. A natural question is whether one can get any hardness results for hypergraph coloring. Indeed, Guruswami et al [8] showed hardness of coloring 2-colorable 4-uniform hypergraphs with constantly many colors. The known hardness results for (hyper)graph coloring are summarized in Table 2.

A question left open by Guruswami et al [8] is whether similar hardness results hold for 3-uniform hypergraphs (as opposed to 4-uniform hypergraphs). We answer this question positively by proving that

Theorem 1.1 *For every constant $\delta > 0$, it is NP-hard to distinguish whether an n-vertex 3-uniform hypergraph is*

	Holds for	**Colors with**
Blum, Karger [4]	3-colorable graphs	$\tilde{O}(n^{3/14})$ colors
Karger et al [13]	k-colorable graphs, $k \geq 3$	$\tilde{O}(n^{1-\frac{3}{k+1}})$ colors; Improvements by Halperin et al [9]
Krivelevich et al [18]	2-colorable 3-uniform hypergraphs	$\tilde{O}(n^{1/5})$ colors

Table 1 : Known algorithmic results for approximate (hyper)graph coloring.

	Holds for	**Hardness of coloring with**	**Assumption**
Khanna et al [14]	3-colorable graphs	4 colors	P $\neq$ NP
Khot [15]	k-colorable graphs, all sufficiently large k	$k^{\frac{1}{25}(\log k)}$ colors	P $\neq$ NP
Guruswami et al [8]	2-colorable 4-uniform hypergraphs	constantly many colors	P $\neq$ NP
Guruswami et al [8]	2-colorable 4-uniform hypergraphs	$\Omega(\frac{\log\log n}{\log\log\log n})$ colors	NP $\not\subseteq$ DTIME($n^{O(\log\log n)}$)
Khot [16]	k-colorable 4-uniform hypergraphs, $k \geq 5$	$(\log n)^{ck}$ colors, $c > 0$ absolute constant	NP $\not\subseteq$ DTIME($2^{(\log n)^{O(1)}}$)
This paper	3-colorable 3-uniform hypergraphs	constantly many colors	P $\neq$ NP
Dinur et al [6]	2-colorable 3-uniform hypergraphs	constantly many colors	P $\neq$ NP

Table 2 : Known hardness results for approximate (hyper)graph coloring.

3-*colorable or it contains no independent set of size* δn. *In particular, it is NP-hard to color* 3-*colorable* 3-*uniform hypergraphs with constantly many colors.*

Independent set in a hypergraph is defined as a set of vertices such that no edge lies entirely within this set. In a properly colored hypergraph, the vertices colored with the same color form an independent set. Thus saying that a hypergraph contains no independent set of size δn is stronger than saying it cannot be colored with $1/\delta$ colors.

Under a stronger complexity assumption than P$\neq$NP, we show the following stronger hardness result.

Theorem 1.2 *Assuming NP* $\not\subseteq$ *DTIME(*$2^{(\log n)^{O(1)}}$*), it is hard to color a* 3-*colorable* 3-*uniform hypergraph on* n *vertices with* $(\log\log n)^c$ *colors. One can take* $c = 1/9$.

Maximization Version : In the maximization version of the hypergraph coloring problem, we are given a q-uniform hypergraph and k different colors. The goal is to assign one color to every vertex so as to maximize the number of edges that are non-monochromatic.

One can also think of the maximization version as a constraint satisfaction problem (CSP) on the Not-All-Equal predicate. The vertices of the hypergraph are variables of the CSP and the edges are constraints of the CSP. Since the hypergraph is q-uniform, every constraint is defined on q variables. Assignment of colors to the vertices corresponds to assigning every variable a value from a domain of size k. A constraint is satisfied iff all its variables are *not* assigned the same value, or equivalently, the corresponding edge is non-monochromatic. We call this CSP to be the problem NAE $_{q,k}$. The optimum of the CSP is the maximum fraction of the constraints that can be satisfied by any assignment.

Note that assigning every variable a random value from the size-k domain satisfies $(1 - \frac{1}{k^{q-1}})$ fraction of the constraints. The problem NAE $_{q,k}$ is said to have a *Random Threshold Property* if it is NP-hard to do strictly better than assigning random values, or more specifically, if it is NP-hard to distinguish whether the optimum is $\geq 1 - \epsilon$ or $\leq 1 - \frac{1}{k^{q-1}} + \epsilon$ for arbitrarily small $\epsilon > 0$.

Håstad [11] showed that for the problem NAE $_{4,2}$, a gap $(1, \frac{7}{8} + \epsilon)$ is hard, i.e. this problem has the random threshold property. This result was the basic starting point for the hypergraph coloring hardness result of Guruswami et al [8]. On the other hand, Zwick [21] showed that for the problem NAE $_{3,2}$, there exists an algorithm that does strictly better than the random assignment and this seemed to be a stumbling block in extending Guruswami et al's techniques to 3-uniform hypergraphs. Khot [17] investigated the case when the domain is ternary, i.e. the problem NAE $_{3,3}$. He showed that a certain conjecture about 2-prover games (which he calls Unique Games Conjecture) implies that the problem NAE $_{3,3}$ has the random threshold property. An open problem was whether such a result could be proven unconditionally. In this paper, we solve this problem by showing that

Theorem 1.3 *For every $k \geq 3$, the problem NAE $_{3,k}$ has the random threshold property on satisfiable instances. In particular, it is NP-hard to distinguish whether a 3-uniform hypergraph is 3-colorable or any coloring of the vertices with 3 colors has at most $\frac{8}{9}+\epsilon$ fraction of the edges non-monochromatic where $\epsilon > 0$ is an arbitrarily small constant.*

Our Techniques : Our results are obtained via new constructions of probabilistically checkable proofs (PCPs) for NP. Most PCP constructions in the recent years ([11], [8] for example) can be divided into two components, the so-called *Outer PCP Verifier* and the so-called *Inner PCP Verifier*. The outer verifier used in all previous constructions is based on the hardness of a 2-prover 1-round game obtained by parallel repetition of a basic 2-prover protocol for Max-3SAT. A crucial property of these games is that the answer of the first prover uniquely determines the answer of the second prover if the verifier is to accept. Let us denote by (q_1, q_2) the pair of questions asked to the first and the second prover respectively. For every answer a of the first prover, let $\pi_{q_1,q_2}(a)$ be the answer of the second prover if the verifier is to accept (this notation makes sense because the answer of the first prover uniquely determines the answer of the second prover).

Khot [17] pointed out that one possible direction for getting hardness results for some open problems (Min-2CNF deletion for example), is to show hardness of 2-prover games with the following additional property : for every question-pair (q_1, q_2), and for every two distinct answers a, a' of the first prover, we have $\pi_{q_1,q_2}(a) \neq \pi_{q_1,q_2}(a')$.

The main technique (which is simple but powerful) in this paper is to get a weaker analogue of the above property. We will show hardness of 2-prover games where the maps π_{q_1,q_2} are "smooth". For every question q_1 to the first prover and every two distinct answers a, a' of the first prover, we will have

$$\Pr_{q_2}\left[\,\pi_{q_1,q_2}(a) \neq \pi_{q_1,q_2}(a')\,\right] \approx 1 (\text{Smoothness property})$$

Thus over the choice of the question q_2 to the second prover (conditional on the question to the first prover being q_1), the answers $\pi_{q_1,q_2}(a)$ and $\pi_{q_1,q_2}(a')$ are distinct with high probability.

We combine this technique with the multi-layered construction of an outer PCP verifier by Dinur, Guruswami, Khot and Regev [5] (which they used to show $q-1-\epsilon$ hardness for vertex cover in q-uniform hypergraphs).

We believe that the techniques in this paper will be quite useful in future. As another application of our techniques, we obtain simpler proofs of Håstad's [11] hardness results for Max-3SAT and Set Splitting on satisfiable instances and Holmerin's [12] hardness result for vertex cover in hypergraphs. We omit these proofs from this paper due to space limitations.

Remark : Dinur, Regev and Smyth [6] obtained a better hardness result for coloring 3-uniform hypergraphs, and our paper was partly influenced by their result. They show hardness of coloring 2-colorable 3-uniform hypergraphs with constantly many colors. However their construction has large independent sets (in fact independent sets of size $\frac{n}{2}$) and in this respect, their result is weaker than Theorem 1.1. Their techniques are quite different from ours. We use a new construction of the PCP outer verifier followed by Fourier analysis whereas Dinur et al make a clever use of Kneser graphs and their construction is more combinatorial in nature. It is interesting that they are able to get a hardness result for 2-colorable 3-uniform hypergraphs even though the problem NAE $_{3,2}$ does *not* have the random threshold property. It would be nice, if possible, to obtain a Fourier analysis based proof of Dinur et al's result.

Overview of the paper : Section 2 introduces some of the tools we use in this paper. In Section 3, we describe the construction of the PCP outer verifier. We prove Theorem 1.1 in Section 4. Theorem 1.3 is proved in Section 5.

2 Preliminaries

2.1. Probabilistic Checking of Proofs

Let $\mathrm{PCP}_{c,s}[r(n), q(n)]$ be the class of languages L for which there exists a probabilistic polynomial time verifier V with the following properties. The verifier has access to a proof Π. On input x with length n, the verifier uses $r(n)$ random bits, queries $q(n)$ bits from the proof Π and then accepts or rejects. It satisfies :
(Completeness condition) :

$$x \in L \Longrightarrow \exists \text{ proof } \Pi \quad s.t. \quad \Pr[\text{ V accepts } \Pi\,] \geq c$$

(Soundness condition) :

$$x \notin L \Longrightarrow \forall \text{ proofs } \Pi, \ \Pr[\text{ V accepts } \Pi\,] \leq s$$

The parameters c and s, $1 \geq c > s > 0$, are called the completeness and soundness parameters respectively.

It was an amazing result by Arora and Safra [2] and Arora et al [1] that the class NP is precisely the class of languages that have a PCP verifier that uses only logarithmically many random bits and a constant number of queries. We state this result below which is known as the PCP Theorem.

Theorem 2.1 (The PCP Theorem, [2], [1])

$$\mathrm{NP} = \mathrm{PCP}_{1,\frac{1}{2}}[O(\log n), O(1)]$$

The PCP Theorem has had a great impact on the theory of inapproximability. Over the past decade, different PCP constructions have led to several hardness results (in many cases optimal results) for various optimization problems, Håstad's results for clique [10] and Max-3SAT [11] and Feige's result for Set-Cover [7] to name a few.

2.2. The Basic Outer Verifier

In this section we will explain the main technique in this paper. We begin with a result on hardness of Max-3SAT.

The PCP Theorem is equivalent to saying that Max-3SAT is MAX-SNP hard. With a suitable transformation, one can show that the hardness holds for 3SAT formulae with a regular structure, i.e. every clause contains exactly 3 variables and every variable appears in exactly 5 clauses (see [7]). We call such formulae to be instances of 3-SAT-5. Thus we have the theorem :

Theorem 2.2 *For some absolute constant $c < 1$, it is NP-hard to distinguish whether a 3-SAT-5 instance is satisfiable (called the YES instance) or there is no assignment satisfying a fraction c of the clauses (called the NO instance).*

Let's first review the standard construction of the outer verifier based on the 3-SAT-5 instance given by the above theorem (see [3], [10], [11]). The construction can be viewed as a 2-prover 1-round game. Let u be a parameter, thought of as a large constant. The verifier picks a set of u clauses at random, say $W = \{C_i \mid i = 1, 2, \ldots, u\}$, and asks the first prover to give a satisfying assignment to these clauses. For every i, the verifier picks a variable x_i at random from the clause C_i. Let $U = \{x_i | i = 1, 2, \ldots, u\}$ be the set of these variables. The verifier asks the second prover to give an assignment to the set U. The verifier accepts iff the answers of the two provers agree on the set of variables U. If we denote $\pi^{U,W}$ to be the projection that maps an assignment to W to its sub-assignment to U, the verifier accepts iff the answer of the first prover is σ and the answer of the second prover is $\pi^{U,W}(\sigma)$.

Clearly, if the 3-SAT-5 formula is satisfiable (i.e. it is a YES instance), the provers have a strategy to make the verifier accept with probability 1 (completeness property). The Parallel Repetition Theorem of Raz [20] gives the soundness property :

Theorem 2.3 *If the 3-SAT-5 instance is a NO instance, then the provers in the above 2-prover 1-round protocol have no strategy that makes the verifier accept with probability more than $2^{-\Omega(u)}$.*

Now we modify this protocol as follows : The verifier picks a set W of u clauses at random and the set U of u variables as before. Then he picks a set W' of Tu clauses at random. He asks the first prover to give a satisfying assignment to the set $W'' = W \cup W'$ and asks the second prover to give a satisfying assignment to the set $U'' = U \cup W'$. The verifier accepts iff the answers of the two provers agree on the set U''.

The clauses in W' are "dummy" in some sense and it can be easily shown that this protocol too has soundness $2^{-\Omega(u)}$. Fix the set W'' of $(T+1)u$ clauses, hence fixing the question asked to the first prover. The question asked to the second prover can be equivalently viewed as picking a random subset $W \subseteq W''$ of u clauses and taking one variable from each clause in W giving a set of variables U. The question to the second prover is $U'' = U \cup (W'' \setminus W)$. Let $\pi^{U'',W''}$ be the projection that maps an assignment to the set W'' to its sub-assignment to the set U''.

The following lemma gives a crucial property of this construction. The significance of this lemma is quite technical in nature and it will be evident only when we get to the actual analysis in Section 4.

Lemma 2.4 (Smoothness property) *For fixed W'' and any two distinct assignments σ_1, σ_2 to W'',*

$$\Pr_{U''}\left[\pi^{U'',W''}(\sigma_1) \neq \pi^{U'',W''}(\sigma_2)\right] \geq 1 - \frac{1}{T}$$

Proof: The assignments σ_1, σ_2 differ on at least one clause $C_0 \in W''$. Note that W is a random subset of W'' with $|W| = u$ and $|W''| = (T+1)u$. Hence with probability $1 - \frac{1}{T}$ over the choice of the set W, $C_0 \notin W$ and consequently $C_0 \in U''$. Whenever this happens, the sub-assignments of σ_1, σ_2 to the set U'' are distinct. ■

We combine this basic technique with the multi-layered construction of an outer PCP verifier by Dinur, Guruswami, Khot and Regev [5]. The combined construction appears in Section 3. A proof for a PCP verifier is obtained by writing down the answers of the provers in the 2-prover protocol. For the sake of convenience, we avoid using the language of 2-prover games, and instead talk in terms of written proofs.

2.3. Long Codes over $GF(k)$ and Fourier Analysis

Long code (over $GF(k)$) over a domain M is indexed by all functions $f : M \mapsto \{1, \omega, \omega^2, \ldots, \omega^{k-1}\}$ where ω is the k^{th} root of unity, i.e. $\omega = e^{2\pi i/k}$. Let

$$\mathcal{F} := \{f \mid f : M \mapsto \{1, \omega, \omega^2, \ldots, \omega^{k-1}\}\}$$

The long code A of $a \in M$ is defined as

$$A(f) = f(a) \quad \forall f \in \mathcal{F}$$

Consider the space of all "tables" $A : \mathcal{F} \mapsto \mathbf{C}$. In particular, a long code is one such table. Consider the characters χ_α where $\alpha : M \mapsto GF(k)$. There is one such character for every α. The character χ_α is a table defined by

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

The characters form an orthonormal basis under the following definition of inner product of tables. For tables A_1, A_2, define

$$< A_1, A_2 > := \frac{1}{k^{|M|}} \sum_{f \in \mathcal{F}} A_1(f)\overline{A_2(f)} = E_f\left[A_1(f)\overline{A_2(f)}\right]$$

It follows that any table can be expanded as $A = \sum_\alpha \widehat{A}_\alpha \chi_\alpha$ where $\widehat{A}_\alpha$ are the Fourier coefficients with $\sum_\alpha |\widehat{A}_\alpha|^2 = < A, A >$. When $A : \mathcal{F} \mapsto \{1, \omega, \ldots, \omega^{k-1}\}$, we have $\sum_\alpha |\widehat{A}_\alpha|^2 = 1$. The Fourier coefficients are given by

$$\widehat{A}_\alpha = < A, \chi_\alpha > = E_f\left[A(f)\overline{\chi_\alpha(f)}\right]$$

In particular when $\alpha \equiv 0$, the value of the coefficient $\widehat{A}_0$ is just $E_f[A(f)]$.

3. A New Outer PCP Verifier

In this section, we explain the construction of the outer PCP verifier from a 3-SAT-5 instance given by Theorem 2.2. The variables will be denoted by $x_1, x_2, \ldots$ and the clauses by $C_1, C_2, \ldots$.

This construction is obtained by combining the technique in Section 2.2 with a construction due to Dinur, Guruswami, Khot and Regev [5]. The construction has parameters (L, T, u). Think of $T \gg L \gg 1$.

Definition 3.1 *For $0 \le i \le L$, a type-i block is defined to be the union of a set of $(L-i)u$ variables and a set of $(T+i)u$ clauses. The family of all type-i blocks is denoted by $\mathcal{V}_i$.*

Remarks : (1) A block is union of a set of variables and a set of clauses. The fact that the components are **sets** and not **tuples** is important. By definition, there is no order associated with elements of a set. (2) The construction of Dinur et al [5] corresponds to the case $T = 0$.

Definition 3.2 *For $0 \le i < j \le L$, we say that a block $U \in \mathcal{V}_i$ is a sub-block of a block $W \in \mathcal{V}_j$ if one can obtain block U by replacing $(j-i)u$ clauses $\{C_l \mid l = 1, 2, .., (j-i)u\}$ in W by $(j-i)u$ variables $\{x_l \mid l = 1, 2, ..., (j-i)u\}$ such that the variable x_l is contained in the clause C_l for $1 \le l \le (j-i)u$.*

For $W \in \mathcal{V}_j$, a random sub-block of W of type-i is obtained by choosing $(j-i)u$ clauses at random from the $(T+j)u$ clauses in W and replacing each clause by one of the variables appearing in that clause picked at random.

A block $U \in \mathcal{V}_i$ contains $(L-i)u$ variables and $(T+i)u$ clauses, and thus a total of $(L-i)u + 3(T+i)u$ variables. A satisfying assignment to the block U is an assignment to these $(L-i)u+3(T+i)u$ variables such that the assignment satisfies all the clauses in U. Let $\mathcal{M}_U$ denote the set of all satisfying assignments to the block U.

If U is a sub-block of W, then every satisfying assignment to W can be restricted to a satisfying assignment to U. Let the map $\pi^{U,W} : \mathcal{M}_W \mapsto \mathcal{M}_U$ denote this operation of taking a sub-assignment/restriction.

Definition 3.3 *We construct a multi-layered graph $\mathcal{G} = \mathcal{G}(L, T, u)$ as follows :*

1. *The graph has $(L+1)$ layers numbered 0 through L and $\mathcal{V}_i$ is the set of vertices in the i^{th} layer. Thus the vertices in the i^{th} layer are the blocks of type-i.*
2. *There are no edges between the vertices of the same layer.*
3. *For $0 \le i < j \le L$, the edges between layers i and j are defined as follows : For a pair of blocks $U \in \mathcal{V}_i$ and $W \in \mathcal{V}_j$ such that U is a sub-block of W, let (U, W) be an edge of the graph.*

Since the 3-SAT-5 instance is regular, the bipartite graph between every pair of layers is regular.

Definition 3.4 *An assignment Φ to the graph $\mathcal{G}(L, T, u)$ assigns a satisfying assignment $\Phi(U) \in \mathcal{M}_U$ to every block U. If U is a sub-block of W, then we say that Φ satisfies an edge (U, W) of the graph if $\Phi(U)$ is a restriction of $\Phi(W)$. In other words*

$$\Phi \text{ satisfies the edge } (U, W) \text{ if } \pi^{U,W}(\Phi(W)) = \Phi(U)$$

Let $OPT(\mathcal{G}, i, j)$ denote the maximum fraction of the edges satisfied between layers i, j by any assignment Φ to these layers.

Dinur et al [5] proved the following theorem. The first property in this theorem follows from Raz's Parallel Repetition Theorem and the second property follows from the regular structure of the graph $\mathcal{G}$. The second property is a weak notion of expansion.

Theorem 3.5 *For every constant $\delta > 0$ and every $L \ge 4/\delta$, the multi-layered graph $\mathcal{G}(L, T, u)$ has the following properties :*

- *If the 3-SAT-5 instance is a YES instance, there exists an assignment Φ that satisfies every edge in the graph $\mathcal{G}$. If the 3-SAT-5 instance is a NO instance, $OPT(\mathcal{G}, i, j) \le 2^{-\Omega((j-i)u)} \le 2^{-\Omega(u)}$ for every pair of layers $i < j$.*

- *Consider any k layers numbered $i_1 < i_2 < \ldots < i_k$ where $k = 4/\delta$. Choose any sets $\mathcal{S}_{i_l} \subseteq \mathcal{V}_{i_l}$ with $|\mathcal{S}_{i_l}| = \delta|\mathcal{V}_{i_l}|$ for $1 \leq l \leq k$. Then there exist two layers numbered i_l and $i_{l'}$ such that the number of edges between the sets $\mathcal{S}_{i_l}$ and $\mathcal{S}_{i_{l'}}$ is at least a fraction $\delta^3/16$ of the total number of edges between the layers i_l and $i_{l'}$.*

Now we show the smoothness of the maps $\pi^{U,W}$, simliar to the Lemma 2.4.

Lemma 3.6 *Let $0 \leq i < j \leq L$ and W be a block of type-j. Let $\sigma, \sigma' \in \mathcal{M}_W$ be two distinct satisfying assignments to W. If U is a random sub-block of W of type i, then*

$$\Pr{}_U\left[\pi^{U,W}(\sigma) = \pi^{U,W}(\sigma')\right] \leq \frac{L}{T}$$

Proof: σ, σ' differ in at least one bit. The projection $\pi^{U,W}$ preserves the variables in W, but replaces some clauses by variables. If σ, σ' differ on a variable in W, their projections under $\pi^{U,W}$ are still distinct. Otherwise they differ on some clause, say clause C_0. For a choice of a random sub-block U, one replaces at random $(j-i)u$ clauses out of the $(T+j)u$ clauses in W. With probability $1 - \frac{j-i}{T+j} \geq 1 - \frac{L}{T}$, the clause C_0 is **not** replaced and hence projections of σ, σ' are distinct. ∎

Lemma 3.7 *Let $0 \leq i < j \leq L$ and W be a block of type-j. Let $\beta \subseteq \mathcal{M}_W$ be non-empty and $\sigma \in \beta$. If U is a random sub-block of W of type-i, then with probability $1 - |\beta|L/T$ we have,*

$$\forall\, \sigma' \in \beta,\ \sigma' \neq \sigma, \quad \pi^{U,W}(\sigma) \neq \pi^{U,W}(\sigma')$$

Proof: Just apply the previous lemma to every $\sigma' \in \beta, \sigma' \neq \sigma$ and take a union bound. ∎

4. Hardness of 3-Uniform Hypergraph Coloring

In this section, we prove Theorem 1.1. We construct a PCP verifier that expects a proof over the domain $GF(3)$. The verifier reads 3 symbols from the proof and accepts iff not all of the 3 symbols are equal. We let the locations in the proof to be the vertices of a hypergraph and the tests of the verifier (reading 3 locations) to be the edges of the hypergraph. This defines a 3-uniform hypergraph. If the 3-SAT-5 instance is a YES instance, there exists a proof that the verifier always accepts and hence the hypergraph is 3-colorable. If the 3-SAT-5 instance is a NO instance, we show that the hypergraph has no independent set of relative size δ.

We expect the proof to contain for every block U, the long code A_U (over $GF(3)$) of a supposed satisfying assignment to U. So this long code is over the domain $\mathcal{M}_U$. Let $\omega = e^{2\pi i/3}$ be the cube root of unity. The long code is indexed by all functions $f \in \mathcal{F}_U$ where

$$\mathcal{F}_U := \{f \mid f : \mathcal{M}_U \mapsto \{1, \omega, \omega^2\}\}$$

The verifier's action is :

- Pick $0 \leq i < j \leq L$ at random.
- Pick a random block W of type-j and its random sub-block U of type-i. Let $\pi^{U,W} : \mathcal{M}_W \mapsto \mathcal{M}_U$ be the projection between U and W. Let $A = A_U$ and $B = A_W$ be the supposed long codes of assignments to the blocks U and W. These long codes are indexed by functions $f \in \mathcal{F}_U$ and $g \in \mathcal{F}_W$ respectively.
- Pick random functions $f \in \mathcal{F}_U, g \in \mathcal{F}_W$. Pick a function $\mu \in \mathcal{F}_W$ by defining for every $x \in \mathcal{M}_W$

$$\mu(x) := \begin{cases} \omega \text{ with probability } 1/2 \\ \omega^2 \text{ with probability } 1/2 \end{cases}$$

- Let $h \in \mathcal{F}_W$ be defined as $h := \overline{g} \cdot \overline{f \circ \pi^{U,W}} \cdot \mu$
- Accept iff

$$\text{Not-All-Equal}(A(f), B(g), B(h))$$

Remark : The above construction of the hypergraph motivates the use of the multi-layered version of the outer PCP verifier. Lets see what happens when the outer verifier has only 2 layers (as is the case with the standard outer verifier used in [8], [11]). Every edge of the hypergraph will have one vertex in the first layer and two vertices in the second layer. Coloring all the vertices in the first layer with one color and all the vertices in the second layer with another color will give a proper 2-coloring of the hypergraph. Thus we won't be able to claim that the hypergraph is not colorable with constantly many colors. However this "cheating strategy" breaks down with the multi-layered construction.

4.1. Completeness

In a correct proof, A will be the long code of some $a \in \mathcal{M}_U$ and B will be the long code of some $b \in \mathcal{M}_W$, with $\pi^{U,W}(b) = a$. In that case

$$A(f) = f(a),\ B(g) = g(b),$$

$$B(h) = h(b) = \overline{g(b) f(\pi^{U,W}(b))}\mu(b) = \overline{g(b) f(a)}\mu(b)$$

Hence $A(f), B(g), B(h)$ can't all be equal, since μ takes values only in the set $\{\omega, \omega^2\}$. Thus the test always accepts a correct proof. Hence the hypergraph is 3-colorable.

4.2. Soundness

We will show that the size of any independent set in the hypergraph is at most δ fraction of the size of the whole hypergraph. The hypergraph is a weighted hypergraph. The vertices in the same layer have equal weight. The total weight of all the vertices in any layer is $\frac{1}{L+1}$. Since there are $L+1$ layers, the total weight of all the vertices in the hypergraph is 1.

Let $\mathcal{I}$ be any set of locations in the proof with weight δ. We will show that there is at least a constant fraction of the tests for which all the 3 queries lie in the set $\mathcal{I}$. This means that for any set of vertices with weight δ in the hypergraph, at least a constant fraction of the edges are contained in this set. Hence there is no independent set of size δ.

Define tables A_U as follows : For every block U,

$$A_U(f) := \begin{cases} 1 \text{ if the location } A_U(f) \in \mathcal{I} \\ 0 \text{ otherwise} \end{cases}$$

Remark : The tables A_U are 0-1 tables whereas the proofs are supposed to be over the alphabet $\{1, \omega, \omega^2\}$. However 0-1 tables still make sense in the Fourier analysis we do and in fact going to 0-1 tables is a trick we exploit.

Call a block U "good" if at least $\delta/2$ fraction of the locations in the table A_U are set to 1. Equivalently, U is good if the Fourier coefficient $\widehat{A}_{U,0}$ of the table A_U satisfies

$$\widehat{A}_{U,0} \ \geq \ \delta/2$$

By an averaging argument, at least $\delta/2$ fraction of the blocks U are good. Again by an averaging argument, in at least $\delta/4$ fraction of the layers, at least $\delta/4$ fraction of the blocks are good. The number of such layers is $\geq \delta/4 \cdot L \geq O(1/\delta)$ provided $L > O(1/\delta^2)$.

By the "weak expansion property" of the multi-layered graph (see Theorem 3.5), there exist two layers $i_0 < j_0$ such that at least $\delta/4$ fraction of the blocks in each of the two layers are good and the number of edges between the good blocks is at least $\delta\,' = \Omega(\delta^3)$ fraction of the total number of edges between the layers i_0 and j_0. Fix these layers i_0 and j_0 for the rest of the proof.

Consider the process of picking a random block W from layer j_0 and its random sub-block of type-i_0. As noted, with probability δ', both W and U are good. Denoting by $A = A_U$ and $B = A_W$, we have

$$E_{U,W}\left[\widehat{A}_0 \widehat{B}_0^2\right] \geq (\delta/2)^3 \delta' = \delta'' \text{ (say)} \tag{1}$$

Now consider the probability that all the three queries $A(f), B(g), B(h)$ lie in the set $\mathcal{I}$. Every proof location is 1 or 0 depending on whether it is in the set $\mathcal{I}$ or not. Hence this probability is the expectation

$$E_{U,W,f,g,\mu}\left[A(f)B(g)B(h)\right]$$

which can be expanded using Fourier expansions of tables A and B as

$$E_{U,W,f,g,\mu}\Big[\sum_{\alpha,\beta,\gamma} \widehat{A}_\alpha \widehat{B}_\beta \widehat{B}_\gamma \, \chi_\alpha(f)\, \chi_\beta(g)\, \chi_\gamma(\overline{g}\,\overline{f \circ \pi}\mu)\Big]$$

where we denoted $\pi = \pi^{U,W}$ for notational convenience. Note that $\beta, \gamma : \mathcal{M}_W \mapsto GF(3)$ and $\alpha : \mathcal{M}_U \mapsto GF(3)$. The expression can be written as

$$E_{U,W,f,g,\mu}\Big[\sum_{\alpha,\beta,\gamma} \widehat{A}_\alpha \widehat{B}_\beta \widehat{B}_\gamma$$

$$\chi_\alpha(f)\, \chi_\beta(g)\, \overline{\chi_\gamma(g)}\, \overline{\chi_\gamma(f \circ \pi)}\, \chi_\gamma(\mu)\Big] =$$

$$E_{U,W,f,g,\mu}\Big[\sum_{\alpha,\beta,\gamma} \widehat{A}_\alpha \widehat{B}_\beta \widehat{B}_\gamma \, \chi_\alpha(f)\, \chi_{\beta-\gamma}(g)\, \overline{\chi_\gamma(f \circ \pi)}\, \chi_\gamma(\mu)\Big]$$

This expectation is zero unless $\beta = \gamma$. Also $\chi_\beta(f \circ \pi) = \chi_{\pi_3(\beta)}(f)$ if we let $\pi_3(\beta) : \mathcal{M}_U \mapsto GF(3)$ be a function defined as

$$\text{For } y \in \mathcal{M}_U, \quad \pi_3(\beta)(y) := \sum_{x \,:\, \pi(x)=y} \beta(x)$$

where the sum is over $GF(3)$). Hence the expression simplifies to

$$E_{U,W,f,g,\mu}\Big[\sum_{\alpha,\beta} \widehat{A}_\alpha \widehat{B}_\beta^2 \, \chi_{\alpha-\pi_3(\beta)}(f)\, \chi_\beta(\mu)\Big]$$

Again, the expression is zero unless $\alpha = \pi_3(\beta)$. Defining "cardinality" of β as

$$|\beta| = |\{x \mid x \in \mathcal{M}_W, \beta(x) \neq 0\}|$$

we get $E_\mu\left[\chi_\beta(\mu)\right] = (-\frac{1}{2})^{|\beta|}$. Thus the expression further simplifies to

$$E_{U,W}\Big[\sum_\beta \widehat{A}_{\pi_3(\beta)} \, \widehat{B}_\beta^2 \, (-\frac{1}{2})^{|\beta|}\Big] \tag{2}$$

Lemma 4.1 *The terms in (2) with $\pi_3(\beta) \neq 0$ can be bounded in magnitude by $2^{-\Omega(u)}$.*

Proof: Using a standard argument, we will show that if the terms with $\pi_3(\beta) \neq 0$ have "high" magnitude, then there exists an assignment Φ to the blocks in layers i_0 and j_0 (see Definition 3.4) which satisfies a "good" fraction of the edges between layers i_0 and j_0. Using Cauchy-Schwartz inequality,

$$\left| E_{U,W}\left[\sum_{\beta:\pi_3(\beta)\neq 0} \widehat{A}_{\pi_3(\beta)} \widehat{B}_\beta^2 (-\frac{1}{2})^{|\beta|}\right]\right|$$

$$\leq E_{U,W}\left[\sqrt{\sum_{\beta:\pi_3(\beta)\neq 0}|\widehat{A}_{\pi_3(\beta)}|^2|\widehat{B}_\beta|^2(\frac{1}{2})^{2|\beta|}}\sqrt{\sum_\beta|\widehat{B}_\beta|^2}\right]$$

$$\leq\sqrt{E_{U,W}\left[\sum_{\beta:\pi_3(\beta)\neq 0}|\widehat{A}_{\pi_3(\beta)}|^2|\widehat{B}_\beta|^2\frac{1}{|\beta|}\right]} \quad (3)$$

Note that A, B are 0-1 tables and thus $\sum_\alpha |\widehat{A}_\alpha|^2 \leq 1$ and $\sum_\beta |\widehat{B}_\beta|^2 \leq 1$. The assignment Φ is defined as follows : For a block W, pick β with probability $|\widehat{B}_\beta|^2$, pick a random $x \in \mathcal{M}_W$ such that $\beta(x) \neq 0$ and define $\Phi(W) = x$. For a block U, pick α with probability $|\widehat{A}_\alpha|^2$, pick a random $y \in \mathcal{M}_U$ such that $\alpha(y) \neq 0$ and define $\Phi(U) = y$. The summation in (3) is precisely the probability that $\pi(\Phi(W)) = \Phi(U)$. This is bounded by $OPT(\mathcal{G}, i_0, j_0) \leq 2^{-\Omega(u)}$ as claimed (see Theorem 3.5). ■

Next, we note that the terms in (2) with $|\beta| \geq \log T$ have magnitude at most $(\frac{1}{2})^{\log T} = \frac{1}{T}$ which can be assumed to be negligible by choosing T large enough.

Thus we are left with terms where $\pi_3(\beta) = 0$ and $|\beta| \leq \log T$. Fix W and consider the case when $\beta \neq 0$, i.e. there exists $x \in \mathcal{M}_W$ such that $\beta(x) \neq 0$. Lemma 3.7 shows that over the choice of a random sub-block U, with probability $1 - \log T \cdot L/T$, we have

$$\forall\, x' \ s.t.\, \beta(x') \neq 0,\ x \neq x' \ \text{ we have } \ \pi(x) \neq \pi(x')$$

Whenever this happens, $\pi_3(\beta) \neq 0$. Hence the probability that $\pi_3(\beta) = 0$ is at most $\frac{\log T \cdot L}{T}$ which is negligible. Thus the terms with $\pi_3(\beta) = 0, \beta \neq 0$ and $|\beta| \leq \log T$ have negligible magnitude.

Finally we are left with the single term

$$E_{U,W}\left[\widehat{A}_0\widehat{B}_0^2\right]$$

which is at least δ'' as observed before (Equation (1)). Assuming the magnitude of all the terms neglected to be at most $\delta''/2$, it follows that at least $\delta''/2$ fraction of the tests between layers i_0 and j_0 have all the 3 queries in the set $\mathcal{I}$. In particular, $\mathcal{I}$ cannot be an independent set. This completes the proof. Theorem 1.2 is proved by plugging in appropriate super-constant values of the parameters L, T, u in the analysis. We omit the details.

5. Hardness of the Problem NAE$_{3,k}$

In this section, we prove Theorem 1.3. We give a proof in the special case when $k \geq 3$ is a prime. The proof can be easily generalized for any odd integer $k \geq 3$. However one requires a somewhat different proof for even k and we omit it from this paper.

We will construct a PCP verifier that reads 3 symbols from a proof over the alphabet $GF(k)$, has perfect completeness and soundness $\frac{k^2-1}{k^2} + \epsilon$ where $\epsilon > 0$ is an arbitrary constant. This clearly suffices to prove Theorem 1.3.

The verifier is based on the construction of the outer verifier in Section 3. However we do not need the "weak expansion property" of this construction (given by property-2 of Theorem 3.5). We make a crucial use of Lemma 3.7.

After picking a block W_j and its sub-block W_i with a suitable distribution, the verifier's test is very similar to the test given in Section 4. Note however that we use long codes over $GF(k)$.

Action of the verifier :

- Pick a random block W_L of type-L. For $i = L, L-1, L-2, \ldots, 1$, let W_{i-1} be a random sub-block of type-$(i-1)$ of the (already chosen) type-i block W_i.
- Let $A_i = A_{W_i}$ be the supposed long code of a satisfying assignment to the block W_i. Let $\mathcal{M}_{W_i}$ be the set of satisfying assignments to the block W_i. For $0 \leq i < j \leq L$, let $\pi^{i,j} = \pi^{W_i,W_j} : \mathcal{M}_{W_j} \mapsto \mathcal{M}_{W_i}$ be the projection function between W_i and W_j.
- Pick $0 \leq i < j \leq L$ at random.
- Pick functions $f : \mathcal{M}_{W_i} \mapsto \{1, \omega, \omega^2, \ldots, \omega^{k-1}\}$, $g : \mathcal{M}_{W_j} \mapsto \{1, \omega, \omega^2, \ldots, \omega^{k-1}\}$ at random.
- Pick a function $\mu : \mathcal{M}_{W_j} \mapsto \{\omega, \omega^2\}$ by defining for every $x \in \mathcal{M}_{W_j}$

$$\mu(x) := \begin{cases} \omega \text{ with probability } 1/2 \\ \omega^2 \text{ with probability } 1/2 \end{cases}$$

- Let $h : \mathcal{M}_{W_j} \mapsto \{1, \omega, \omega^2, \ldots, \omega^{k-1}\}$ be defined as $h := g^2 \cdot \overline{f \circ \pi^{i,j}} \cdot \mu$
- Accept iff

$$\text{Not-all-equal}(A_i(f), A_j(g), A_j(h))$$

5.1. Completeness

Clearly, the test always accepts a correct proof constructed from a satisfying assignment to the 3-SAT-5 instance using correct long codes.

5.2. Soundness

We will show that if the 3-SAT-5 instance is a NO instance, the test accepts with probability at most $\frac{k^2-1}{k^2} + \epsilon$.

Lemma 5.1 *If $x, y, z \in \{1, \omega, \omega^2, \ldots, \omega^{k-1}\}$, then the expression*

$$1 - \frac{1}{k^2} \sum_{r_1+r_2+r_3=0} x^{r_1} y^{r_2} z^{r_3} \qquad (r_1, r_2, r_3 \in GF(k))$$

is 0 *if* $x = y = z$ *and* 1 *otherwise.*

From this lemma, it is clear that the acceptance probability of the verifier is

$$1 - \frac{1}{k^2} \sum_{r_1+r_2+r_3=0} E_{\substack{W_L, W_{L-1}, \ldots, W_0, \\ i,j,f,g,\mu}} \left[A_i(f)^{r_1} A_j(g)^{r_2} A_j(h)^{r_3}\right]$$

Fix $W_L, W_{L-1}, \ldots, W_0, i, j$ for the time being and consider the expectation over f, g, μ. For notational convenience, we will drop the $E[]$ notation. Expanding the tables $A_i^{r_1}, A_j^{r_2}, A_j^{r_3}$ by their Fourier expansions, we get

$$1 - \frac{1}{k^2} \sum_{r_1+r_2+r_3=0} \sum_{\alpha,\gamma,\beta} \widehat{A}^{r_1}_{i,\alpha} \widehat{A}^{r_2}_{j,\gamma} \widehat{A}^{r_3}_{j,\beta}$$

$$\cdot \chi_\alpha(f) \chi_\gamma(g) \chi_\beta(g^2 \overline{f} \mu)$$

The expectation over g is zero unless $\gamma + 2\beta = 0$. Let $\pi = \pi^{i,j}$ and let $\pi_k(\beta)$ be a function $\pi_k(\beta) : \mathcal{M}_{W_i} \mapsto GF(k)$ defined as

$$\forall\, y \in \mathcal{M}_{W_i}, \;\; \pi_k(\beta)(y) := \sum_{x \in \mathcal{M}_{W_j} : \pi(x) = y} \beta(x)$$

With this definition, the expectation over f is zero unless $\alpha = \pi_k(\beta)$. Thus the acceptance probability is

$$1 - \frac{1}{k^2} \sum_{r_1+r_2+r_3=0} \sum_{\beta} \widehat{A}^{r_1}_{i,\pi_k(\beta)} \widehat{A}^{r_2}_{j,-2\beta} \widehat{A}^{r_3}_{j,\beta} \cdot E_\mu[\chi_\beta(\mu)]$$

It can be proved in a similar way as [16, Lemma 8] that

$$E_\mu[\chi_\beta(\mu)] = (1 - \Omega(\frac{1}{k^2}))^{|\beta|} \cdot \mathcal{Q}(\beta)$$

where $\mathcal{Q}(\beta)$ is a complex number with absolute value 1. Therefore the acceptance probability is

$$1 - \frac{1}{k^2} \sum_{r_1+r_2+r_3=0} \sum_{\beta} \widehat{A}^{r_1}_{i,\pi_k(\beta)} \widehat{A}^{r_2}_{j,-2\beta} \widehat{A}^{r_3}_{j,\beta} \cdot$$

$$(1 - \Omega(\frac{1}{k^2}))^{|\beta|} \cdot \mathcal{Q}(\beta)$$

Using Lemma 4.1, the terms with $\pi_k(\beta) \neq 0$ can be used to define a "good" assignment to the blocks in layers i and j. Therefore these terms can be assumed to be arbitrarily small in magnitude. The terms with $|\beta| \geq O(k^2 \log T)$ have magnitude at most $\frac{1}{T}$ which is negligible. Now consider the terms with $\beta \neq 0, \pi_k(\beta) = 0$ and $|\beta| \leq O(k^2 \log T)$. Using Lemma 3.7, for a fixed W_j, over the choice of a random sub-block W_i, the probability that $\pi_k(\beta) = 0$ is at most $\frac{O(k^2 \log T) \cdot L}{T}$ which is negligible. Hence these terms can also be ignored.

Thus we are left with only those terms where $\beta = 0$. The acceptance probability is

$$\Pr[\text{acc}] = 1 - \frac{1}{k^2} \sum_{r_1+r_2+r_3=0} \widehat{A}^{r_1}_{i,0} \widehat{A}^{r_2}_{j,0} \widehat{A}^{r_3}_{j,0} + \epsilon' \qquad (4)$$

where ϵ' takes into account the terms neglected. Let

$$\widehat{A}_{i,0} = \sum_{l=0}^{k-1} a_{il} \cdot \omega^l$$

where a_{il} are non-negative real numbers with $\sum_{l=0}^{k-1} a_{il} = 1$. Note that a_{il} is just the fraction of entries in the table A_i which are equal to ω^l. It follows that

$$\widehat{A}^r_{i,0} = \sum_{l=0}^{k-1} a_{il} \cdot \omega^{rl}$$

Hence the summation in (4) can be written as

$$\sum_{r_1+r_2+r_3=0} \sum_{l_1,l_2,l_3} a_{il_1} a_{jl_2} a_{jl_3} \omega^{r_1 l_1 + r_2 l_2 + r_3 l_3}$$

$$= \sum_{l_1,l_2,l_3} a_{il_1} a_{jl_2} a_{jl_3} \sum_{r_1+r_2+r_3=0} \omega^{r_1 l_1 + r_2 l_2 + r_3 l_3}$$

The inner summation is 0 unless $l_1 = l_2 = l_3$ and k^2 otherwise. So the sum reduces to

$$k^2 \sum_{l=0}^{k-1} a_{il} a_{jl}^2$$

Recall that the verifier first picks blocks $W_L, W_{L-1}, \ldots, W_0$ and then picks i, j at random. Taking expectation over the choice of $0 \leq i, j \leq L$ and applying lemma 5.2, we get

$$\Pr[\text{acc}] = 1 - E_{\,0 \leq i < j \leq L}\left[\sum_{l=0}^{k-1} a_{il} a_{jl}^2\right] + \epsilon'$$

$$\leq 1 - \frac{1}{k^2} + \frac{2}{L} + \epsilon'$$

Lemma 5.2 *Let $\{a_{il} : 0 \leq i \leq L;\; 0 \leq l \leq k-1\}$ be a collection of non-negative reals such that*

$$\sum_{l=0}^{k-1} a_{il} = 1 \quad \forall\, i$$

Then we have

$$E_{\,0 \leq i < j \leq L}\left[\sum_{l=0}^{k-1} a_{il}\; a_{jl}^2\right] \geq \frac{1}{k^2} - \frac{2}{L}$$

Proof: Let $a_{il} = (1+b_{il})/k$ so that $\sum_{l=0}^{k-1} b_{il} = 0$ for every i. Also $-1 \le b_{il} \le k-1$.

$$a_{il}a_{jl}^2 = \frac{1}{k^2}a_{il}(1+b_{jl})^2 = \frac{1}{k^2}a_{il}(1+2b_{jl}+b_{jl}^2)$$
$$\ge \frac{1}{k^2}a_{il}(1+2b_{jl}) = \frac{1}{k^3}(1+b_{il})(1+2b_{jl}) =$$
$$\frac{1}{k^3}(1+b_{il}+2b_{jl}+2b_{il}b_{jl})$$

Summing over l and noting that $\sum_{l=0}^{k-1} b_{il} = 0$, we get

$$\sum_{l=0}^{k-1} a_{il}a_{jl}^2 \ge \frac{1}{k^2} + \frac{2}{k^3}\sum_{l=0}^{k-1} b_{il}b_{jl}.$$ Finally

$$E_{0\le i<j\le L}[b_{il}b_{jl}] = \frac{1}{2\binom{L+1}{2}}\Big((\sum_{i=0}^{L} b_{il})^2 - \sum_{i=0}^{L} b_{il}^2\Big)$$
$$\ge \frac{1}{2\binom{L+1}{2}}(-(L+1)(k-1)^2) \ge -\frac{k^2}{L}$$

■

6. Acknowledgments

I sincerely thank Irit Dinur, Oded Regev and Cliff Smyth for showing their hypergraph coloring result to me. Many thanks to Venkatesan Guruswami for several helpful discussions on the subject.

References

[1] S. Arora, C. Lund, R. Motawani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM*, 45(3):501–555, 1998.

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

[3] M. Bellare, O. Goldreich, and M. Sudan. Free bits, pcps and non-approximability. *Electronic Colloquium on Computational Complexity, Technical Report TR95-024*, 1995.

[4] A. Blum and D. Karger. An $\tilde{O}(n^{3/14})$ coloring algorithm for 3-colorable graphs. *Information Processing Letters*, 61:49–53, 1997.

[5] I. Dinur, V. Guruswami, S. Khot and O. Regev. $k-1-\epsilon$ hardness for vertex cover in k-uniform hypergraphs. In preparation.

[6] I. Dinur, O. Regev and C. Smyth. The hardness of 3-uniform hypergraph coloring. Appearing in *Proc. of the 43rd Annual IEEE Symposium on Foundations of Computer Science*, 2002.

[7] U. Feige. A threshold of ln n for approximating set cover. *Journal of the ACM*, 45(4), 634–652, July 1998.

[8] V. Guruswami, J. Håstad, and M. Sudan. Hardness of approximate hypergraph coloring. In *Proc. of the 41st IEEE Symposium on Foundations of Computer Science*, pages 149–158, 2000. (The references are to the journal version of this paper which is in preparation).

[9] E. Halperin, R. Nathaniel and U. Zwick. Coloring k-colorable graphs using smaller palettes. In *Proc. of the 12th Annual Symposium on Discrete Algorithms*, 319–326, 2001.

[10] J. Håstad. Clique is hard to approximate within $n^{1-\epsilon}$. In *Proc. of the 37th Annual IEEE Symposium on Foundations of Computer Science*, pages 627–636, 1996.

[11] J. Håstad. Some optimal inapproximability results. In *Proc. of the 29th Annual ACM Symposium on Theory of Computing*, pages 1–10, 1997.

[12] J. Holmerin. Vertex cover on 4-regular hyper-graphs is hard to approximate within $2-\epsilon$. To appear in *Proc. of the 34th Annual ACM Symposium on Theory of Computing*, 2002.

[13] D. Karger, R. Motwani, and M. Sudan. Approximate graph coloring by semidefinite programming. In *Proc. of the 35th IEEE Symposium on Foundations of Computer Science*, 1994.

[14] S. Khanna, N. Linial and S. Safra. On the hardness of approximating the chromatic number. In *Proc. of the 2nd Israel Symposium on Theory and Computing Systems*, ISTCS, 250–260, IEEE Computer Society Press, 1993.

[15] S. Khot. Improved inapproximability results for maxclique, chromatic number and approximate graph coloring. In *Proc. of the 42nd IEEE Symposium on Foundations of Computer Science*, 2001.

[16] S. Khot. Hardness results for approximate hypergraph coloring. To appear in *Proc. of the 34th Annual ACM Symposium on Theory of Computing*, 2002.

[17] S. Khot. On the power of unique 2-prover 1-round games. To appear in *Proc. of the 34th Annual ACM Symposium on Theory of Computing*, 2002.

[18] M. Krivelevich, R. Nathaniel and B. Sudakov. Approximating coloring and maximum independent set in 3-uniform hypergraphs. In *Proc. of the 12th Annual Symposium on Discrete Algorithms*, 2001.

[19] L. Lovász. Coverings and colorings of hypergraphs. In *Proc. of 4th Southeastern Conf. on Combinatorics, Graph Theory and Computing*, pages 3–12. Utilitas Mathematica Publishing, Winnipeg, 1973.

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

[21] U. Zwick. Finding almost satisfying assignments. In *Proc. of the 30th Annual ACM Symposium on Theory of Computing*, pages 551–560, 1998.