Witnesses for non-satisfiability of dense random 3CNF formulas

Uriel Feige
Microsoft Research
and Weizmann Institute
urifeige@microsoft.com

Jeong Han Kim
Microsoft Research
Redmond, Washington
jehkim@microsoft.com

Eran Ofek
Weizmann Institute
Rehovot, Israel
eran.ofek@weizmann.ac.il

Abstract

We consider random 3CNF formulas with n variables and m clauses. It is well known that when $m > cn$ (for a sufficiently large constant c), most formulas are not satisfiable. However, it is not known whether such formulas are likely to have polynomial size witnesses that certify that they are not satisfiable. A value of $m \simeq n^{3/2}$ was the forefront of our knowledge in this respect. When $m > cn^{3/2}$, such witnesses are known to exist, based on spectral techniques. When $m < n^{3/2-\epsilon}$, it is known that resolution (which is a common approach for refutation) cannot produce witnesses of size smaller than 2^{n^ϵ}. Likewise, it is known that certain variants of the spectral techniques do not work in this range.

In the current paper we show that when $m > cn^{7/5}$, almost all 3CNF formulas have polynomial size witnesses for non-satisfiability. We also show that such a witness can be found in time $2^{O(n^{0.2}\log n)}$, whenever it exists. Our approach is based on an extension of the known spectral techniques, and involves analyzing a certain fractional packing problem for random 3-uniform hypergraphs.

1. Introduction

The 3SAT problem, namely, deciding whether a 3CNF formula is satisfiable, is one of the central NP-complete problems. Deciding whether a formula is not satisfiable is complete for co-NP. The question of whether there is a nondeterministic polynomial time algorithm that recognizes non-satisfiable 3CNF formulas (or in other words, polynomial size witnesses for non-satisfiability) is equivalent to the well known "NP=co-NP?" open question. The question of whether there are deterministic polynomial time algorithms for non-satisfiability is the even better known "P=co-NP?" question (which is equivalent to the "P=NP?" question). In this paper (similar to many other previous papers, some of which will be mentioned shortly), we study "average case" versions of these questions.

The average case model involves a density parameter β. We consider 3CNF formulas with n variables and $m = \beta n$ clauses, in which the clauses are chosen independently at random. Our results are not sensitive to minor variations on the model, but for concreteness, assume the following model: one takes a random permutation on all possible $2^3\binom{n}{3}$ 3CNF clauses, and picks the first m clauses in the permutation. It is well known (see for example [13, 19]) that when β is sufficiently large (say, $\beta > 5$), almost all such formulas are not satisfiable. For random 3CNF formulas of sufficiently high density β, we consider two tasks:

1. Deterministic refutation. Design a polynomial time algorithm that never accepts a satisfiable formula, and show that it accepts most formulas of density β.
2. Nondeterministic refutation. Design a nondeterministic polynomial time algorithm with properties as above. Equivalently, design polynomial size *witnesses for non-satisfiability* that can be checked (though not necessarily found) in polynomial time, never exist for satisfiable formulas, and exist for most formulas of density β.

Clearly, deterministic refutation is at least as hard as nondeterministic refutation. Observe also that in the context of random formulas, the larger β is, the easier is the refutation task, because any refutation algorithm that applies to smaller densities may simply be run on a prefix of the larger density formula.

Random 3CNF formulas often serve to show the limitations of well known refutation algorithms. Taking *resolution* as a prominent example, it was shown [6, 3, 5, 4] that resolution fails to provide nondeterministic polynomial time refutation for most formulas of densities $n^{1/2-\epsilon}$ (where $\epsilon > 0$ can be taken to be arbitrarily small), though it is known to provide nondeterministic polynomial time refutation for most formulas of densities $n/\log n$. (In resolution, one attempts to refute a CNF formula by deriving the empty clause. The clause $Y \vee Z$ can be derived from the two clauses $Y \vee x$ and $Z \vee \bar{x}$. For more details, see the aforementioned references.)

The strongest refutation algorithms known for random 3SAT are based on a ”spectral” approach first suggested in [17] for 4SAT. This approach involves computing eigenvalues of certain matrices derived from the formula. These spectral algorithms were extended to apply to 3SAT [14, 18, 11], and they are known to deterministically refute most formulas of density $cn^{1/2}$, for a sufficiently large constant c. Attempts to extend these techniques so as to work for densities below $\sqrt{n}$ have failed so far. For example, in [12] it is shown that a natural use of semidefinite programming cannot refute 3CNF formulas at such low densities.

Further motivation for studying deterministic refutation of random 3CNF formulas is given in [9]. There it is shown that if there is no deterministic refutation for most 3CNF formulas with cn clauses (where c is an arbitrarily large constant) then certain combinatorial optimization problems (such as minimum graph bisection and dense k-subgraph) do not have polynomial time approximation schemes. It is an open question whether it is NP-hard to approximate these problems arbitrarily well, though further evidence that these problems are indeed hard to approximate is given in [20].

The main result of this paper is nondeterministic refutation of random 3CNF formulas of densities below $\sqrt{n}$. Our method (which is based on the spectral approach) works already for densities $\beta = cn^{2/5}$ (for sufficiently large $c > 0$), a range where resolution is known not to be polynomial. We do not know whether the witnesses implied by our nondeterministic refutation can be found in polynomial time, but we show that they can be found in time roughly exponential in n/β^2. The authors are not aware of any previous refutation algorithm (whether deterministic or nondeterministic) that was proved to run in time better than exponential in n/β (for random 3CNF formulas of density $\beta < n^{1/2}$).

In an intuitive sense (which we do not wish to make exact), our state of knowledge following the current work is that with respect to the ”average” complexity of 3SAT, co-NP $\subset$ NP for densities above $n^{2/5}$, and co-NP $\subset$ P for densities above $n^{1/2}$. Pushing either of these densities down is an interesting question.

1.1. The main idea

Given previous work on refutation, the main new idea that we introduce is fairly simple, though apparently is was not previously observed. Proving that this idea actually works requires some nontrivial probabilistic analysis.

The starting point is a principle that was explicitly introduced in [9], and later used in refutation algorithms, such as in [11]. The principle relates between satisfying assignments of random formulas, and assignments that satisfy most clauses as if they were 3XOR clauses, namely, set an odd number of literals (either one or three) to true.

Proposition 1.1 *There is a polynomial time algorithm that for almost every 3CNF formula of density β proves that every satisfying assignment must satisfy all but at most $c\sqrt{\beta}n$ clauses as 3XOR, where c is some universal constant.*

Proposition 1.1 is given for the sake of intuition, but the detailed presentation of our approach will not refer to it explicitly. For this reason we shall not present the proof of Proposition 1.1, but only note that it follows by combining the first part of the proof of Theorem 2.6 with the proofs of Lemmas 3.2 and 3.3.

The new aspect of our work is in the use of inconsistent tuples of clauses.

Definition 1.2 *A collection of k clauses is an* even k-tuple *if every variable appears in it an even number of times. (Observe that the fact that we are dealing with 3CNF clauses implies that k must be even.) An even k-tuple is an* inconsistent k-tuple *if the total number of appearances of negated literals in its clauses is odd (and hence this also holds for positive literals).*

The significance of inconsistent tuples of clauses comes from the following proposition.

Proposition 1.3 *For any assignment to the variables, at least one of the clauses of an inconsistent k-tuple is not satisfied as 3-XOR.*

Proof: View every clause (ℓ_1, ℓ_2, ℓ_3) as an equation $\ell_1 + \ell_2 + \ell_3 = 1$ modulo 2. Assigning a literal ℓ_i to true (false, respectively) will be interpreted as setting $\ell_i = 1$ ($\ell_i = 0$, respectively). An assignment satisfies the clause as 3XOR iff it satisfies the corresponding equation. Pick an arbitrary assignment and substitute the corresponding values in the equations of the k-tuple. Summing up all equations, the right hand side gives 0 (modulo 2), because k is even. The left hand side gives 1 (modulo 2). This can be seen as follows. If all literals were positive, than for any assignment, the left hand side sums up to 0 (modulo 2), because there is an even number of occurrences of each literal. Flipping a single literal flips the sum modulo 2. As there is an odd number of negative literals in an inconsistent k-tuple, the sum must be 1 (modulo 2). Having established that the left hand side of the sum differs from the right hand side, we can deduce that at least one equation is not satisfied, and hence at least one clause is not satisfied as 3XOR. □

A random 3CNF formula is expected to contain many inconsistent k-tuples, if k is sufficiently large. For example, it is not hard to prove the following lemma.

Lemma 1.4 *If $k\beta^2 \gg n$, then the expected number of inconsistent k-tuples in a random 3CNF formula (expectation taken over choice of formula) is ”large” (say, more than m).*

Lemma 1.4 by itself does not suffice for our refutation algorithm, and is given merely for intuition. Its proof is omitted.

In its simplest form, our witness of non-satisfiability will be composed of $t > c\sqrt{\beta}n$ disjoint inconsistent k-tuples. On the one hand, Proposition 1.3 implies that in any assignment, at least t different clauses are not satisfied as 3XOR. On the other hand, Proposition 1.1 implies that in any satisfying assignment, there are at most $c\sqrt{\beta}n$ of clauses not satisfied as 3XOR. The condition $t > c\sqrt{\beta}n$ implies that no satisfying assignment can exist.

If k-tuples are to be disjoint, then necessarily $t \leq m/k$, and thus $m/k \geq n\sqrt{\beta}$, implying $k < \sqrt{\beta}$. Together with the condition $k\beta^2 > n$ of Lemma 1.4, this implies that our approach can potentially work when $\beta > n^{2/5}$.

The description above is an oversimplification of our refutation approach. It turns out to be advantageous to allow some limited overlap between inconsistent k-tuples, and compensate for this by taking a larger number of inconsistent k-tuples. On a conceptual level, having more flexibility allows our approach to be applied to a wider range of formulas. But more important for the current context, there are concrete technical reasons why allowing overlap is desirable. One reason is that it is not clear to us whether known techniques suffice in order to prove that there are $\Omega(n^{1.2})$ disjoint inconsistent k-tuples (for $k \simeq n^{0.2}$) in a random 3CNF formula with $m \simeq n^{1.4}$ clauses. Intuitively, the source of the difficulty is as follows. Once one packs $\Omega(n)$ k-tuples, this uses up $\Omega(n^{1.2})$ clauses, and a new "random" k-tuple is likely to hit one of these clauses, and hence not to be disjoint from the existing k-tuples. By allowing overlap between k-tuples, this source of difficulty is avoided. Thereafter, the probabilistic analysis needed in order to prove that our approach works becomes manageable (though it still remains complicated and requires expertise in the probabilistic method). Another reason for allowing overlap concerns the design of algorithms for finding the witness for non-satisfiability. Having a witness that depends on a disjoint collection of k-tuples spells bad news, because (disjoint) set packing problems are notoriously difficult to solve (and are also NP-hard to approximate within a factor of $O(k^{1-\epsilon})$). In contrast, once we allow overlap between sets, the underlying computational problem resembles a fractional packing problem, and these problems can be handled more efficiently. This will be used in the proof of Theorem 4.1.

2. The witness for non-satisfiability

In this section we present the components of the witness for non-satifiability, and prove that no satisfiable 3CNF formula can possibly contain such a witness (and hence the witness proves non-satifiability). The input formula will be denoted by ϕ.

The first two components of our witness were used also in previous work on refuting random 3CNF formulas [9, 11].

Definition 2.1 *Let ϕ be a 3CNF formula with n variables and m clauses. The* imbalance *of a variable i (denoted by I_i) is the difference in absolute value between the number of times it appears with positive polarity and the number of times it appears with negative polarity. The* total imbalance *of ϕ is $I_\phi = \sum_{i=1}^{n} I_i$.*

The first component of our witness is I_ϕ, the imbalance of ϕ. The smaller I_ϕ is, the better.

Definition 2.2 *Let ϕ be a 3-CNF formula with n variables (denoted by $x_1, \ldots, x_n$) and m clauses. The* matrix induced by ϕ *is a symmetric matrix of order n that will be denoted by M_ϕ. Its entries are derived from ϕ as follows. Initially, all entries are 0. Thereafter, every clause of ϕ changes six of the entries, two entries for each pair of variables in the clause. The change is $+1/2$ if the polarities of the variables do not match, and $-1/2$ if they do match. For example, the clause $(x_i, x_j, \bar{x}_k)$ changes M_{jk} (and M_{kj}, to preserve symmetry) and M_{ik} (and M_{ki}) by $+1/2$, and M_{ij} (and M_{ji}) by $-1/2$.*

The second component of our witness is the largest eigenvalue of M_ϕ, which we shall denote by λ. The smaller the absolute value of λ is, the better. The use of eigenvalues as part of refutation algorithms for CNF formulas was introduced in [17] and used in several works thereafter.

Remark. We assume for simplicity of the presentation that λ (which might not be rational) can be represented efficiently with infinite precision. A more formal treatment may replace λ everywhere in this manuscript either by $\lambda+1$ rounded to the nearest integer (when λ is very close to being an integer), or by $\lceil \lambda \rceil$ otherwise. Details are omitted.

Recall the notion of an inconsistent tuple from Definition 1.2.

Definition 2.3 *A* (k, t, d)-collection *is a collection of t inconsistent k-tuples, in which every inconsistent k-tuple contains only clauses from ϕ, and every clause from ϕ is contained in at most d of the inconsistent k-tuples.*

The third component of our witness is a (k, t, d)-collection. This component will be most effective when the ratio t/d is large. Note that necessarily $tk \leq md$, and hence to have t/d large, we need k to be small.

We now present a complete description of our witness for non-satisfiability.

The witness. Given a 3CNF formula ϕ with n variables and m clauses, the witness is composed of the following three components:

1. The value I_ϕ of the imbalance, as defined in Definition 2.1.
2. The largest eigenvalue λ of the matrix M_ϕ that was defined in Definition 2.2.
3. A (k, t, d)-collection as defined in Definition 2.3, with $t < n^2$.

The components need to satisfy the following **refutation inequality**:

$$t > \frac{d(I_\phi + \lambda n)}{2}. \qquad (1)$$

This completes the description of the witness.

The condition $t < n^2$ is imposed only so as to ensure that the witness is of polynomial size, and serves no other purpose. Likewise, the exact value of k is not important, though clearly $k \leq m$. Moreover, it does not matter whether all inconsistent tuples in the collection have the same cardinality k, but we assume they do, so as to simplify the presentation in this paper.

Proposition 2.4 *The witness can be checked in polynomial time.*

Proof: The imbalance I_ϕ can be computed in polynomial time, and hence can be checked in polynomial time. The same applies to the eigenvalue λ (see also the remark following Definition 2.2). In fact, neither I_ϕ nor λ need to be given explicitly as part of the witness, as both can be computed efficiently from ϕ.

For every even k-tuple in the (k, t, d) collection, one needs to check that every variable appears in the respective clauses an even number of times, that every clause indeed belongs to ϕ, and that the number of negative literals is indeed odd. Moreover, one needs to check that every clause appears in at most d of the even k-tuples, and that the total number of even k-tuples is t. Clearly, all these checks can be made in polynomial time (in n, m), because $t < n^2$.

The refutation inequality can also be checked in polynomial time. (Again, see also remark following Definition 2.2.) □

We now show that a satisfiable formula cannot have a witness as described above. We first present a known connection between the eigenvalue λ of M_ϕ and assignments that satisfy clauses of ϕ in a ”not all equal” (NAE) fashion, namely, satisfy either one or two literals in a clause.

Lemma 2.5 *If there is an assignment that satisfies m_1 clauses in ϕ as NAE, then the largest eigenvalue λ of M_ϕ is at least $(4m_1 - 3m)/n$.*

Proof: Let A be an assignment that satisfies m_1 clauses of ϕ as NAE. Consider the n-dimensional vector v_A that has value 1 on coordinates corresponding to variables that A sets to true, and value -1 on coordinates corresponding to variables that A sets to false. The fact that v_A has norm $\sqrt{n}$ implies that $n\lambda \geq v_A^t M_\phi v_A$. Using the definition of M_ϕ, it is not hard to see that every clause that is satisfied either once or twice by A contributes $+1$ to $v_A^t M_\phi v_A$, whereas every other clause contributes -3. Hence $n\lambda \geq m_1 - 3(m - m_1) = 4m_1 - 3m$. □

Observe that a random assignment satisfies $3m/4$ clauses as NAE in expectation, and hence the lower bound on λ implied by Lemma 2.5 is nonnegative.

Theorem 2.6 *Let ϕ be a 3CNF formula with n variables and m clauses. If ϕ has a witness as described above that satisfies the refutation inequality (1), then there is no assignment satisfying ϕ.*

Proof: Assume for the sake of contradiction that ϕ is satisfiable, and let A be a satisfying assignment. By definition of the notion of imbalance, A can satisfy at most $(3m + I_\phi)/2$ literals. Every clause contains at least one of these satisfied literals. Lemma 2.5 implies that A satisfies at most $m_1 = (3m + \lambda n)/4$ clauses as NAE. The rest of the $m - m_1$ clause must be satisfied three times by A. Hence each of them contains two more satisfied literals (beyond the one already counted). It follows that the number of clauses containing two literals satisfied by A is at most:

$$\frac{3m + I_\phi}{2} - m - 2(m - m_1) = -\frac{3m}{2} + \frac{I_\phi}{2} + 2m_1 \leq \frac{I_\phi + \lambda n}{2}$$

Hence A satisfies at least $m - (I_\phi + \lambda n)/2$ as 3XOR (this relates to the 3XOR principle of Proposition 1.1).

We now turn to the (k, t, d) collection. By Proposition 1.3, each of the t inconsistent k-tuples must contain at least one clause not satisfied as 3XOR by A. As there are at most $(I_\phi + \lambda n)/2$ such clauses, and each of them participates in at most d inconsistent k-tuples, there can be a satisfying assignment A only if $t \leq d(I_\phi + \lambda n)/2$. □

3. Dense random 3CNF formulas have witnesses

In section 2 we presented a witness certifying that a 3CNF formula is not satisfiable. In this section, we show that most 3CNF formulas with $m \gg n^{1.4}$ clauses have such a witness.

Theorem 3.1 *Let ϕ be a random 3CNF formula with n variables and $m = \beta n$ clauses, where $\beta = cn^{0.4}$ for a sufficiently large constant c. Then almost surely:*

1. *The imbalance satisfies $I_\phi = O(n\sqrt{\beta}) = O(n^{1.2})$.*
2. *The largest eigenvalue satisfies $\lambda = O(\sqrt{\beta}) = O(n^{0.2})$.*

3. *There are (k,t,d) collections with parameters $k = O(n/\beta^2) = O(n^{0.2})$, $t = \Omega(n\beta) = \Omega(n^{1.4})$ and $d = O(k) = O(n^{0.2})$.*

Items 1 and 2 above are known (being part of the 3XOR principle), and their proofs are presented in Lemmas 3.2 and 3.3. The more challenging part of our analysis is to prove item 3, and the proof is given in Section 3.4. Section 3.1 explains how different ingredients of the proofs fit together.

Substituting these parameters in the refutation inequality (1), we see that the left hand side is $\Omega(n\beta)$, whereas the right hand side is $O(n^2/\beta^{3/2})$. Hence the inequality is satisfied when $\beta > cn^{2/5}$, for some sufficiently large constant c.

3.1. General observations

The most complicated part of our proof is to prove the existence of (k,t,d) collections. To simplify the presentation of this part of the proof, we shall fix $\beta = n^{0.4}$ (which is the smallest value that interests us), and prove that $d = O(n^{0.2})$ and $t = \Omega(n^{1.4})$, without insisting that the value of the leading constants is such that the refutation inequality is satisfied. However, this suffices for our purpose for the following reason. Increasing β by some constant factor c, increases I_ϕ and λ by $O(\sqrt{c})$, d can be kept fixed, and then t is increased by a factor of c (by treating the random formula as a concatenation of c random formulas). Hence regardless of the leading constants in the O and Ω notation, we can choose c sufficiently large so as to make the refutation inequality hold.

As ϕ is chosen at random, the three parameters I_ϕ, λ and the maximum possible t (for a given fixed d) are random variables. All three random variables enjoy the bounded difference property. Namely, adding one clause to ϕ can change I_ϕ by at most 3, change λ by at most 1 (because the matrix associated with a single clause has no eigenvalue whose absolute value is larger than 1), and change t by at most d (once we have fixed d). As a consequence of this, it can be shown that all these random variables are highly concentrated around their means or medians (see for example [1]). Hence it suffices to show that the medians or means of I_ϕ, λ and the maximum possible t are in the desired range, and this will imply that the fraction of ϕ that have a witness is overwhelming (at least $1 - O(2^{-n^\delta})$ for some $\delta > 0$).

The strong concentration results also imply that we may use interchangeably whichever is more convenient of the common models for generating random formulas. For example, we may use a model in which exactly m clauses are chosen at random, and consider either the version with or without replacement. Alternatively, we may consider a model in which each of the possible $M = 2^3\binom{n}{3}$ clauses is chosen to be in ϕ independently with probability m/M. Another variation is a model in which we choose each 3-tuple of variables to be a clause independently with probability $m/\binom{n}{3}$, and thereafter choose the polarities of the variables independently at random. All these models are sufficiently similar to each other (say, when $m < n^{3/2}$) so that the events that we consider happen with overwhelming probability in one of the models iff they happen with overwhelming probability in all models. (See for example [21] for a similar setting.)

We omit the formal proofs of the observations made above in this section.

3.2. The imbalance

The following lemma is known and its proof is given here for completeness.

Lemma 3.2 *The expected imbalance (over the choice of 3CNF formula ϕ with n variables and $m = \beta n > n$ clauses) satisfies $E(I_\phi) = O(n\sqrt{\beta})$.*

Proof: All expectations and probabilities in this proof are taken over the choice of ϕ.

For any variable x_i we denote by d_i the number of appearances of x in ϕ_1. It holds that $\sum_{i=1}^n E[d_i] = 3m$. By symmetry, for every i it holds that $E[d_i] = \frac{3m}{n}$, which we denote by d (for the purpose of this proof, not to be confused with the parameter d used elsewhere). Given that $d_i = j$ the polarities of the appearances of x_i are random and independent. Hence $E[I_i^2 \mid d_i = j] = j$. It then follows that

$$E[I_i^2] = \sum_k \Pr[d_i = j]\; E[I_i^2 \mid d_i = j]$$

$$= \sum_j j \Pr[d_i = j] = E[d_i] = d.$$

Using the convexity of the square function

$$E[I_i] \le \sqrt{E[I_i^2]} \le \sqrt{d}.$$

By linearity of expectation, $E[\sum_{i=1}^n I_i] \le n\sqrt{d}$. In particular, with probability at least $1/2$, $I_\phi \le 4n\sqrt{\beta}$. □

3.3. The largest eigenvalue

The following lemma is known and its proof is sketched for completeness.

Lemma 3.3 *The value λ of the largest eigenvalue of M_ϕ satisfies $\lambda = O(\sqrt{\beta})$ with high probability (over the choice of 3CNF formula ϕ with n variables and $m = \beta n$ clauses, and using for simplicity the choice $\beta \ge n^{2/5}$).*

Proof: (sketch) Consider the model in which each of the $2^3\binom{n}{3}$ possible clauses is chosen to be in ϕ independently at random with probability $m/8\binom{n}{3}$. Each clause contains three pairs of variables. View the matrix M_ϕ as the sum of three matrices, $M_1 + M_2 + M_3$, where each of these matrices involves the contributions of just one type of pair (e.g., M_1 contains the contributions of the pairs composed of the first and second variables in every clause of ϕ). For every matrix separately, its entries are identically distributed and statistically independent (except for the constraints imposed by symmetry). Each entry is distributed symmetrically around 0, and has variance $O(\beta/n)$. The results of [16] then imply that with high probability, the largest eigenvalue of each of the three matrices is at most $O(\sqrt{\beta})$. Clearly, λ cannot be larger than the sum of these three eigenvalues. □

Remark. Lemma 3.3 is incorrect when $\beta \ll \log n / \log \log n$. However, variations of it can be extended all the way down to constant values of β. See [10] for details.

3.4. Collections of inconsistent tuples

In this section we show that a random 3CNF formula with $n^{1.4}$ clauses is likely to have a (k, t, d) collection with $k = O(n^{0.2})$, $t = \Omega(n^{1.4})$ and $d = O(n^{0.2})$.

It is more convenient to first find a collection of even k-tuples with the above parameters, and only later extract from it those even k-tuples that are also inconsistent. When considering even k-tuples, the polarity of variables does not matter. Hence a clause can be viewed as a 3-tuple of variables. In this case, a 3CNF formula can be viewed as a 3-uniform hypergraph over n vertices, where every clause corresponds to a hyperedge. A natural model for random 3-uniform hypergraphs is one in which each of the hyperedges is inserted independently with probability p. This hypergraph can model a 3CNF formula in which each 3-tuple of variables forms a clause with probability p, and thereafter the polarities of variables are set independently at random. In our context, the appropriate value for the parameter p is $n^{-1.6}$, as this corresponds to a formula with $\Theta(n^{1.4})$ clauses (in expectation). A 2-regular subhypergraph induced by k hyperedges (namely, a collection of k hyperedges in which every vertex appears either twice or not at all) corresponds to an even k-tuple of clauses. (Even k-tuples are somewhat more general in the sense that variables can appear any even number of times, but we shall not need this generality here.) The most complicated technical part of this manuscript is the proof of the following theorem.

Theorem 3.4 *A 3-uniform hypergraph with n vertices in which every possible hyperedge is included independently with probability $p = n^{-1.6}$ is likely to contain a collection of $t = \Omega(n^{1.4})$ 2-regular subhypergraphs such that every vertex participates in at most $d = O(n^{0.2})$ of these subhypergraphs.*

The proof of this theorem is deferred to section 3.5, though the proof of some of the related lemmas will be omitted due to space limitations. (A note concerning notation: in Section 3.5, k will denote the number of vertices in the 2-regular subhypergraph, rather than the number of hyperedges.) Here we sketch the overall structure of the proof.

It is relatively easy to prove that k can be chosen to have some value close to $n^{0.2}$, in a way that causes the expected number of (not necessarily induced) 2-regular subhypergraphs with k vertices to be roughly $n^{1.4}$. However, large expectation does not automatically mean a high probability event. For example, in the context of random 3CNF formulas, it is known that at density $\beta = 5$, the expected number of satisfying assignments of a random 3CNF formula is exponentially large, but still almost all such formulas are not satisfiable [19]. To turn expectation results into high probability results, one may try to bound the variance.

To allow us to bound the variance, we exclude some of the 2-regular subhypergraphs from consideration. The excluded 2-regular subypergraphs are those that include subcollections of hyperedges that are ”dense”, namely involve relatively few vertices compared to the number of hyperedges. As a simple example, we do not wish to allow a 2-regular subhypergraph to contain two hyperedges that share two vertices (and hence contain only four vertices in total). More generally, for every value of ℓ, we require a certain lower bound on the number of vertices that every subcollection of ℓ hyperedges needs to contain. 2-regular subhypergraphs that meet these requirements for all values of ℓ will be called *expanding*. The reason to concentrate on 2-regular expanding subhypergraphs with no dense subcollections is that dense subcollections are correlated with the existence of 2-regular subhypergraphs (which themselves are dense – they have a ratio of 3:2 between variables and clauses), and hence dense subcollections have large effect on the variance.

We show that a random 2-regular subhypergraph has constant (though a small constant) probability of being expanding. Hence the expected number of expanding 2-regular subhypergraphs is still $\Theta(n^{1.4})$. Now detailed calculations show that the variance is small, and so with high probability the actual number is also $\Omega(n^{1.4})$.

It remains to show that d, the number of 2-regular hypergraphs in which a hyperedge may participate, is small, $O(n^{0.2})$. Again, it is not hard to show that in expectation this is the case, but as explained before, expectation by itself does not suffice. To avoid tedious variance calculations, we now restrict the structure of the collection of 2-regular subhypergraphs: we allow every two subhypergraphs to share at most one hyperedge. A relatively easy computation based on expectations shows that this does not decrease the size of the collection by much. But now, for every hyperedge, all 2-regular subhypergraphs in the collection that contain it share no other hyperedge with each other. This eliminates positive correlations that they might have had, and allows us to prove that with high probability their number is as expected.

Corollary 3.5 *Let ϕ be a random 3CNF formula with n variables and $n^{1.4}/8$ clauses. Then with high probability ϕ contains a (k, t, d) collection with parameters $t = \Omega(n^{1.4})$ and $d = O(n^{0.2})$.*

Proof:(sketch) Theorem 3.4 implies that with high probability ϕ contains an even collection with the above parameters. As the polarities are random, a symmetry argument implies that with probability $1/2$, at least half of the even k-tuples are inconsistent. This shows that the corollary holds with constant probability. As hinted in section 3.1, Talagrand's inequality can be used to boost this probability up (essentially to $1 - e^{-n^{0.8}}$, details omitted). □

3.5. Random 3-uniform hypergraphs

A hypergraph is 3-uniform if every hyperedge contains exactly three vertices, and 2-regular if every vertex is contained in exactly two hyperedges. Let $H(n, p)$ be a random 3-uniform hypergraph on the set V of n vertices in which each 3-tuple $\{x, y, z\}$ of distinct vertices becomes an (hyper)edge with probability $p = n^{-1.6}$, independently of all others. In this section we show that (with high probability) $H(n, p)$ contains $\Theta(n^{1.4})$ 2-regular subhypergraphs on $k = \Theta(n^{0.2})$ vertices such that each edge in $H(n, p)$ is in $O(n^{0.2})$ of those subhypergraphs.

3.5.1 Properties of the random 2-regular 3-uniform hypergraph

For k divisible by 3, a perfect 3-matching of $2k$ elements is a decomposition of the $2k$ elements into $2k/3$ sets of size 3. To generate the 2-regular 3-uniform hypergraph $H_2(k\,;3)$ uniformly at random among all simple 2-regular 3-uniform hypergraphs on $k = \Theta(n^{0.2})$ vertices, one may use the configuration model (see [21] and references therein, for example). For each vertex v, take two copies of v, say v', v''. The copies v', v'' are called clones of v. The clones are used to generate $H_2(k\,;3)$. Provided k (and hence $2k$) is divisible by 3, generate a uniform random perfect 3-matching of all clones. The edges of the hypergraph induced by the perfect 3-matching can be obtained by contracting both of v' and v'' into v. That is, $\{u, v, w\}$ is an edge if and only if the perfect 3-matching contains a 3-tuple consisting of clones of u, v, w. In general, the induced hypergraph $H_2^*(k\,;3)$ may have loops and/or multiple edges, where a loop is an edge containing a vertex twice or more like $\{u, u, w\}$. Calling a hypergraph without loops and multiple edges *simple*, it is known [7] that the random hypergraph is simple with probability $(1 + o(1))e^{-1}$. It is also easy to check that each simple 2-regular 3-uniform hypergraph on k vertices arises from exactly 2^k perfect 3-matchings of clones. Therefore, the induced random hypergraph conditioned on being simple yields the uniform random 2-regular 3-uniform hypergraph $H_2(k\,;3)$.

Moreover, the uniform random perfect 3-matching may be generated by selecting a uniform random permutation of $2k$ clones so that each perfect 3-matching may be realized by precisely $(3!)^{2k/3}(2k/3)!$ permutations. In particular, there are

$$\alpha(2k) := \frac{(2k)!}{6^{2k/3}(\frac{2k}{3})!} = \frac{(1 + o(1))3^{1/2}}{(4\pi k)^{1/3}} 2^{-2k/3}((2k)!)^{2/3} \tag{2}$$

perfect 3-matchings.

The hypergraph $H_2^*(k\,;3)$ induced by the configuration enjoys nice expansion properties. We characterize some of these properties in terms of the maximum number $\beta(\ell)$ of vertices of degree 2 in a subhypergraph with ℓ edges of $H_2(k, 3)$. As k will be large (some increasing function of n) in our intended applications, the following lemma addresses only the case that k is sufficiently large (the lemma is trivially incorrect when $k \leq 40$).

Lemma 3.6 *For some fixed constant $\delta > 0$, for every sufficiently large k, the induced random hypergraph $H_2^*(k\,;3)$ satisfies the following with probability at least δ:*

$$\beta(\ell) \leq \begin{cases} \ell - 1 & \textit{for } 1 \leq \ell \leq 20 \\ 1.1\ell & \textit{for } 21 \leq \ell \leq \frac{k}{\log k} \\ 1.41\ell & \textit{for } \frac{k}{\log k} \leq \ell \leq \frac{k}{3}, \end{cases}$$

especially, $H_2^*(k\,;3)$ is a simple hypergraph.

For a set L of edges in a 3-uniform hypergraph, let $V(L)$ be the set of all vertices contained in edges in L, and $V_j(L)$ is the set of vertices contained in precisely j edges in L, $j \geq 1$. If L can be extended to a 2-regular 3-uniform hypergraph with the expansion properties described in Lemma 3.6, then $3|L| = |V_1(L)| + 2|V_2(L)|$ and $|V(L)| = |V_1(L)| + |V_2(L)|$ imply that

$$|V(L)| = 3|L| - |V_2(L)| \geq 3|L| - \beta(|L|). \tag{3}$$

Corollary 3.7 *If L can be extended to a 2-regular 3-uniform hypergraph with the expansion properties described in Lemma 3.6, then, for $\ell = |L|$,*

$$|V(L)| \geq \begin{cases} 2\ell + 1 & \textit{for } 1 \leq \ell \leq 20 \\ 1.9\ell & \textit{for } 21 \leq \ell \leq \frac{k}{\log k} \\ 1.59\ell & \textit{for } \frac{k}{\log k} \leq \ell \leq \frac{k}{3}. \end{cases}$$

In particular, L induces a simple hypergraph.

3.5.2 Many 2-regular subhypergraphs

Our task in this section is to show that $H(n,p)$ contains a collection of $\Theta(n^{1.4})$ 2-regular subhypergraphs. To show this, we shall limit our attention only to 2-regular subhypergraphs that have expansion properties as in Lemma 3.6, because these expansion properties will be used in certain variance calculations.

It is not hard to see that k can be chosen such that the expected number of expanding $H_2(k;3)$ is as desired. Indeed, Lemma 3.6 implies that, in expectation, there are

$$\Theta\Big(\binom{n}{k}2^{-k}\alpha(2k)p^{2k/3}\Big)$$

2-regular subgraphs of $H(n,p)$ with the expansion properties described in the lemma. Observing

$$\binom{n}{k}2^{-k}\alpha(2k)p^{2k/3} = \Theta\Big(\frac{n^k((2k)!)^{2/3}}{k^{1/3}2^{5k/3}k!n^{3.2k/3}}\Big)$$

$$= \Theta\Big(\frac{1}{k^{1/2}}\Big(\frac{k}{2en^{0.2}}\Big)^{k/3}\Big),$$

we take $k = 2en^{0.2} + 4.5\log n - c_1$ for appropriate constant c_1 so that the mean μ is between $cn^{1.4}/2$ and $cn^{1.4}$ for a constant c determined later. (It may not be possible to take c_1 so that $\mu = (1+o(1))cn^{1.4}$, since k must be an integer.)

Having established that the expected number of $H_2(k;3)$ is as desired, the following lemma uses the second moment method to show that with high probability, the expectation is attained.

Lemma 3.8 *There is $c > 0$ so that, with probability $1 - o(1)$, the random hypergraph $H(n,p)$ with $p = n^{-1.6}$ has a collection of 2-regular subhypergraphs on k vertices satisfying the followings.*

(i) The collection has more than $(1-o(1))\mu$ elements.
(ii) All subhypergraphs in the collection satisfy the expansion properties described in Lemma 3.6.

Proof: Let $H_1, H_2, ...,$ be all the 2-regular hypergraphs on k vertices in V with the expansion properties. Then the expected number of such hypergraphs in $H(n,p)$ is μ. In other words, for the indicator random variable $X_i = 1(H_i \in H(n,p))$, and $X := \sum_{i\geq 1} X_i$, the mean of X is μ.

To compute the variance of X and other related quantities, an estimation in a general setting is convenient. For a set of L edges, we will estimate the mean of $X_L := \sum_{i:L\subseteq H_i} X_i$ conditioned on $L \in H(n,p)$. When $|L| = 1$ and using q to denote the probability that a random $H_2^*(k\,;3)$ has expansion properties, we have

$$\begin{aligned} &E[X_L|L \in H(n,p)] \\ &= \binom{n}{k-3}2^{-(k-3)}\alpha(2k-3)p^{2k/3-1}q \\ &= \frac{(1+o(1))\mu\ \binom{n}{k-3}2^{-(k-3)}\alpha(2k-3)p^{2k/3-1}}{\binom{n}{k}2^{-k}\alpha(2k)p^{2k/3}} \\ &= (1+o(1))\mu k^3n^{-3}2^3(2^{-1}(2k)^2)^{-1}p^{-1} \\ &= (4+o(1))\mu kn^{-1.4}. \end{aligned}$$

Generally, we have the following lemma.

Lemma 3.9 *Let $\ell := |L|$. Then for $\ell = 1$,*

$$E[X_L|L \in H(n,p)] = (4+o(1))\mu kn^{-1.4}$$

and for connected L with $2 \leq \ell \leq 20$

$$E[X_L|L \in H(n,p)] = O(\mu n^{-0.4\ell-0.8}).$$

Otherwise, for $2 \leq \ell \leq 20$,

$$\binom{2k/3}{\ell}E[X_L|L \in H(n,p)] = O(\mu n^{-0.2\ell-1.6}),$$

for $21 \leq \ell \leq k/3$,

$$\binom{2k/3}{\ell}E[X_L|L \in H(n,p)] = O(\mu n^{-(0.07+o(1))\ell}).$$

Similarly, for $k/3 \leq \ell \leq 2k/3-1$,

$$\binom{2k/3}{\ell}E[X_L|L \in H(n,p)] = O(n^{-(0.07+o(1))(2k/3-\ell)}).$$

Using Lemma 3.9 we complete the proof of Lemma 3.8. To estimate the variance of $X = \sum_{i\geq 1} X_i$, consider

$$\sum_{j:j\neq i} \Pr[X_j = 1|X_i = 1] - \Pr[X_j = 1]$$

$$\leq \sum_{\ell\geq 1}\ \sum_{\substack{L:L\subseteq H_i\\ |L|=\ell}} E[X_L|L \in H(n,p)].$$

Lemma 3.9 yields that the case $\ell = 1$ contributes much more than all other cases combined. (Here the distinction between connected and disconnected L is not necessary. The distinction will be needed to prove Lemma 3.11.) Using the fact that any one of the $2k/3$ clauses associated with X_i can be the clause shared with X_j, one obtains

$$E[(X-\mu)^2] = (8/3+o(1))k^2n^{-1.4}\mu^2 = O(n^{1.8}).$$

Chebyschev's Inequality then gives

$$\Pr[|X-\mu| \geq bn^{0.9}] = O(b^{-2}), \quad \text{for } b \geq 1.$$

□

3.5.3 Hyperedges participate in only few 2-regular subhypergraphs

Lemma 3.8 established that with high probability, $H(n,p)$ has collection of roughly $\mu = \Theta(n^{1.4})$ 2-regular subhypergraphs, where each such subhypergraph is on $k = \Theta(n^{0.2})$ vertices and has the expansion properties of Lemma 3.6. It remains to show that we can find a subcollection of this collection that has the additional property that no hyperedge participates in more than $O(k)$ of the subhypergraph. For this purpose we impose an additional constraint on the collection.

Definition 3.10 *A collection of* 2*-regular hypergraphs is called* nearly disjoint *if each pair of hypergraphs in the collection shares at most one hyperedge.*

We can now strengthen Lemma 3.8 so as to achieve our main lemma.

Lemma 3.11 *(Main Lemma) There is $c > 0$ so that, with probability $0.9+o(1)$, the random hypergraph $H(n,p)$ with $p = n^{-1.6}$ has a nearly disjoint collection of* 2*-regular subhypergraphs on k vertices satisfying the following.*

(i) The collection has more than $(1+o(1))\mu/2$ elements.
(ii) All subhypergraphs in the collection satisfy the expansion properties described in Lemma 3.6
(iii) For each edge e in $H(n,p)$, the number of hypergraphs containing e in the collection is at most $(12+o(1))n^{-1.4}\mu k$.

4. Algorithms for finding witnesses

Our witnesses for non-satisfiability are of polynomial size, and they can be checked in polynomial time. In this section we address the question of how such a witness can be found. Observe that our results for sufficiently dense random 3CNF formulas imply not only that witnesses exist, but moreover, that the refutation inequality (1) is satisfied with some slackness. For concreteness, let us call the inequality $t > d(I_\phi + \lambda n)$ the *robust refutation inequality*, and call witnesses for which this inequality hold *robust witnesses*.

Theorem 4.1 *If a 3CNF formula ϕ has a robust witness with $d \gg \log m$, then a witness of non-satisfiability for ϕ can be found in time polynomial in $\binom{m}{k}$. (Recall that m is the number of clauses, and k and d are the respective parameters of the (k,t,d)-collection associated with the robust witness.)*

Proof: (sketch) As noted in Proposition 2.4, computing the imbalance I_ϕ and λ does not pose a problem (especially as it suffices to compute λ only approximately, due to the slackness in the robust witness). The remaining task is to find a (k,t,d)-collection as implied by the robust witness. We may assume that the values of k and d are known (as there are only polynomially many possible values, and all of them can be tried out). Fixing k and d, we propose the following algorithm for finding a (k,d,t)-collection with large t.

First, enumerate all inconsistent k-tuples. Using exhaustive search, this takes time proportional to $\binom{m}{k}$. (We do not know whether there are substantially faster algorithms for finding even a single inconsistent k-tuple.) Let $T_1, T_2, \ldots, T_\ell$ be the list of all inconsistent k-tuples.

Next, set up the following linear program. With every T_i we associate a variable x_i and the constraint $0 \le x_i \le 1$. In addition, for every clause C we have the constraint $\sum_{i|C\in T_i} x_i \le d$. The objective function is to maximize $\sum_i x_i$. The optimal value of the LP is at least t, because the (k,d,t)-collection associated with the robust witness is a solution to the LP. The LP, which has $\ell \le \binom{m}{k}$ variables, can be solved in time polynomial in its size. (It is conceptually simplest to use a generic linear programming algorithm for this purpose, though other options also exist.) For every i, let x_i^* be the (fractional) value given to x_i by the LP.

Now use randomized rounding. Every T_i is chosen into the collection with probability x_i^*. The expected size of the collection is then at least t. Standard concentration results imply that with high probability the size of the collection is at least $t(1-o(1))$, and moreover, that no clause is used more than $d(1+o(1))$ times in the collection. This last fact uses the assumption that $d \gg \log m$. This provides a witness of non-satisfiability, because of the slackness involved in the original robust witness.

Finally, observe that there is no need to actually perform the randomized rounding. The fractional solution to the LP by itself certifies the existence of a $(k, t(1-o(1)), d(1+o(1)))$-collection (as proved by the randomized rounding argument). Hence the refutation algorithm is deterministic rather than randomized. □

Corollary 4.2 *For sufficiently large c, most random 3CNF formulas with n variables and $cn^{1.4}$ clauses can be refuted in time $2^{O(n^{0.2}\log n)}$.*

Proof: The probabilistic analysis of Section 3 implies that most 3CNF formulas of density as in the corollary have robust witnesses with $k = O(n^{0.2})$. The corollary now follows from Theorem 4.1. □

An interesting question is what is the smallest density for which most random 3CNF formulas can be refuted in polynomial time. The best previous bound is $cn^{1.5}$ clauses for some specific constant $c > 1$ [11]. Combining the approach of the current paper with that of [11], we can extend this to arbitrarily small constant $c > 0$. This requires some additional work, but details are omitted from this conference version.

Acknowledgements

This work was supported in part by grants from the German Israeli Foundation (GIF) and the Israeli Science Foundation (ISF). The third author would like to thank Asaf Nussboim for useful discussions.

References

[1] N. Alon, M. Krivelevich and V. H. Vu. *On the concetration of eigenvalues of random symmetric matrices.* Israel J. Math. 131 (2002), 259–267.

[2] N. Alon and J. Spencer. *The Probabilistic Method, 2nd ed.*. Wiley-interscience, New York, NY, 2000.

[3] P. Beame, R. Karp, T. Pitassi and M. Saks. On the complexity of unsatisfiability proofs for random k-CNF formulas. In *Proceedings of 30th STOC, 561–571, 1998.*

[4] E. Ben-Sasson. Expansion in proof complexity. *PhD thesis. Computer Science Department, Hebrew University, Jerusalem, Israel.* 2001.

[5] E. Ben-Sasson and A. Wigderson. Short proofs are narrow-resolution made simple. *Journal of the ACM (JACM)*, 48(2):149–169, 2001.

[6] V. Chvatal and E. Szemeredi. Many hard examples for resolution. *Journal of the ACM*, 35(4):759–768, Oct 1988.

[7] C. Cooper, A. Frieze, M. Molloy, and B. Reed. Perfect Matchings in Random r-Regular, s-Uniform Hypergraphs, *Combinatorics, Probability and Computing* **5** (1996), 1-14.

[8] P. Erdős and P. Tetali, Representations of integers as the sum of k terms, *Random Structures & Algorithms*, **1**, (1990), 245-261.

[9] U. Feige. Relations between average case complexity and approximation complexity. In *Proc. of the 34th Annual ACM Symposium on Theory of Computing*, pages 534–543, 2002.

[10] U. Feige and E. Ofek. Spectral techniques applied to sparse random graphs. *Random Structures and Algorithms*, 27(2):251–275, 2005.

[11] U. Feige and E. Ofek. Easily refutable subformulas of large random 3CNF formulas. In *proceedings of ICALP 2004, 519–530.*

[12] U. Feige and E. Ofek. Random 3CNF formulas elude the Lovasz theta function. *Technical Report MCS06-01, Computer Science and Applied Mathematics, Weizmann Institute of Science, 2006.*

[13] E. Friedgut and J. Bourgain. Sharp thresholds of graph properties, and the k-sat problem. *JAMS: Journal of the American Mathematical Society*, 12(4):1017–1054, 1999.

[14] J. Friedman, A. Goerdt, and M. Krivelevich. Recognizing more unsatisfiable random 3-sat instances efficiently. Technical report, 2003.

[15] J. Friedman, A. Goerdt, and M. Krivelevich. Recognizing more unsatisfiable random k-SAT instances efficiently. *SIAM Journal on Computing*, 35(2): 408–430, 2005.

[16] Z. Furedi and J. Komlos. The Eigenvalues of Random Symmetric Matrices. *Combinatorica, 1 (3) 233-241, 1981.*

[17] A. Goerdt and M. Krivelevich. Efficient recognition of random unsatisfiable k-SAT instances by spectral methods. In *STACS: Annual Symposium on Theoretical Aspects of Computer Science*, pages 294–304, 2001.

[18] A. Goerdt and A. Lanka. Recognizing more random 3-sat instances efficiently. Manuscript, 2003.

[19] S. Janson, Y. C. Stamatiou, and M. Vamvakari. Bounding the unsatisfiability threshold of random 3-sat. *Random Structures and Algorithms*, 17(2):103–116, 2000.

[20] S. Khot. Ruling Out PTAS for Graph Min-Bisection, Densest Subgraph and Bipartite Clique. FOCS 2004, 136–145.

[21] N. Wormald. Models of random regular graphs, *Surveys in Combinatorics*, 1999, J.D. Lamb and D.A. Preece, eds. London Mathematical Society Lecture Note Series, vol 276, pp. 239–298. Cambridge University Press, Cambridge, 1999.