On smoothed k-CNF formulas and the Walksat algorithm

Amin Coja-Oghlan * Uriel Feige† Alan Frieze‡ Michael Krivelevich §
Dan Vilenchik¶

October 5, 2008

Abstract

In this paper we study the model of ε-smoothed k-CNF formulas. Starting from an arbitrary instance F with n variables and $m = dn$ clauses, apply the ε-smoothing operation of flipping the polarity of every literal in every clause independently at random with probability ε. Keeping ε and k fixed, and letting the density $d = m/n$ grow, it is rather easy to see that for $d \geq \varepsilon^{-k} \ln 2$, F becomes *whp* unsatisfiable after smoothing.

We show that a lower density that behaves roughly like ε^{-k+1} suffices for this purpose. We also show that our bound on d is nearly best possible in the sense that there are k-CNF formulas F of slightly lower density that *whp* remain satisfiable after smoothing.

One consequence of our proof is a new lower bound of $\Omega(2^k/k^2)$ on the density up to which `Walksat` solves random k-CNFs in polynomial time *whp*. We are not aware of any previous rigorous analysis showing that `Walksat` is successful at densities that are increasing as a function of k.

*University of Edinburgh . E-mail: acoghlan@inf.ed.ac.uk.

†The Weizmann Institute. E-mail: uriel.feige@weizmann.ac.il. Supported in part by The Israel Science Foundation (grant No. 873/08)

‡Carnegie Mellon University. E-mail: alan@random.math.cmu.edu. Supported in part by NSF grant DMS0753472

§Tel-Aviv University. E-mail: krivelev@post.tau.ac.il. Research supported in part by a USA-Israel BSF Grant, by a grant from the Israel Science Foundation, and by Pazy Memorial Award.

¶Tel-Aviv University. E-mail: vilenchi@post.tau.ac.il.

1 Introduction

In trying to understand the inherent hardness of the k-satisfiability problem, many researchers analyzed structural properties of formulas drawn from different distributions. One such natural distribution is the following: fix $c, n > 0$ (c may depend on n), choose $m = cn$ clauses uniformly at random out of $2^k\binom{n}{k}$ possible ones. We denote this distribution by $\mathcal{F}_{n,m,k}$. Despite its simplicity, many essential properties of this model are yet to be understood. In particular, the hardness of deciding if a random formula is satisfiable, and finding a satisfying assignment for a random formula, are both major open problems [7, 15].

A remarkable phenomenon occurring in the random model $\mathcal{F}_{n,m,k}$ is a phase transition with respect to the property of being satisfiable. More precisely, there exists a threshold $d_k = d_k(n)$ such that a k-CNF formula with clause-variable ratio greater than d_k is not satisfiable *whp*[1], while one with ratio smaller than d_k is [11].

As a warm up, let us consider the following very simple calculation that provides an upper bound on d_k. Let X_m be a random variable counting the number of satisfying assignments that a random formula F (drawn from $\mathcal{F}_{n,m,k}$) has. It is not hard to see that

$$E[X_m] \leq 2^n \cdot \left(1 - 2^{-k}\right)^m. \tag{1.1}$$

Fix ψ, and ask what is the probability that it satisfies F. Every clause is satisfied with probability $1 - 2^{-k}$, and there are m clauses (the correlation between them is negative). Finally, use the linearity of expectation. One can verify that $E[X_m] = o(1)$ for $m/n \geq 2^k \ln 2$, thus upper bounding d_k (by Markov's inequality).

Observe that this calculation would have been correct in the following model as well: choose an arbitrary k-CNF on n variables and m clauses with only true literals. Negate each literal in each clause with probability $1/2$. If the k-CNF is not arbitrary, but m tuples are chosen uniformly at random out of $\binom{n}{k}$ possible ones, then the resulting distribution is statistically close to $\mathcal{F}_{n,m,k}$ (for $m/n = O(1)$). Indeed for $\mathcal{F}_{n,m,k}$ the upper bound on the satisfiability threshold given in (1.1) is tight as k grows ([2] gave a lower bound of $2^k \ln 2 - k/2 - O(1)$). One interesting question is to find a necessary and sufficient condition for the bound to be tight on the new model we introduced.

In this paper we study a question of similar flavor using the following model for generating random k-CNF formulas.

> *Given a formula F, flip the polarity of every literal in every clause, independently of the others, with probability ε.*

For a formula F, we let F^* be the ε-perturbation of F just described. Let $d_{\varepsilon,k}$ be the minimal number s.t. for every k-CNF F with at least $d_{\varepsilon,k}n$ clauses, F^* is *whp* not satisfiable.

Remark 1. Observe that $d_{\varepsilon,k} = d_{1-\varepsilon,k}$ since for every F with dn clauses we can consider $\bar{F}$ which is identical to F just that every literal in every clause is flipped. The ε-perturbation of F is distributed exactly like the $(1-\varepsilon)$-perturbation of $\bar{F}$. Therefore throughout our discussion we assume w.l.o.g. that $\varepsilon \leq 1/2$.

[1] Writing *whp* we mean with probability tending to 1 as n goes to infinity.

The following question naturally arises after having defined the new model:

Question 2. *What is the (asymptotic) dependency of $d_{\varepsilon,k}$ on ε and k?*

In this paper we give (up to factors of logarithmic order) the dependency of $d_{\varepsilon,k}$ on ε and k. The answer to this question is far from being trivial, and even "guessing" the right dependency is somewhat tricky as we shall shortly explain.

The model that we study was first studied by Feige in the context of refutation [8]. In [9], it is proven that random 3CNF formulas with $cn^{3/2}$ clauses, c some sufficiently large constant, can be refuted in polynomial time. Then, in [8] it was shown that for every F with at least $\varepsilon^{-2}n^{3/2}(\log\log n)^{1/2}$ clauses, F^* can be refuted *whp*.

Our model is also part of the Smoothed Analysis paradigm which was introduced by Spielman and Teng in [18] to help explain why the simplex algorithm for linear programming works well in practice but not in (worst-case) theory. They considered instances formed by taking an arbitrary constraint matrix and perturbing it by adding independent Gaussian noise with variance ε to each entry. They showed that, in this case, the shadow-vertex pivot rule succeeds in expected polynomial time. Our work joins a long series of papers published since [18] studying perturbed instances in a variety of problems. In this context one can also mention the work in [14] who studied another model for smoothing k-CNF formulas (the smoothing operation is adding random clauses).

1.1 Our contribution

In this paper we establish upper and lower bounds on the threshold $d_{\varepsilon,k}$. Here is a first try for an upper bound on $d_{\varepsilon,k}$, which is probably the natural go at the problem, but as it turns out is not the right answer.

Proposition 3. *Let F be any k-CNF instance with m clauses over n variables, and let $d = m/n$. Let ε be the perturbation parameter, and let F^* be the perturbed instance. If $d > \varepsilon^{-k}\ln 2$ then whp F^* is not satisfiable.*

The proof of this proposition is a simple first moment calculation. Fix an assignment ψ, and consider a clause C in F. If C contains exactly i literals which are true under ψ, then the ε-smoothing of C is satisfied by ψ with probability $1-(1-\varepsilon)^{k-i}\varepsilon^i \leq 1-\varepsilon^k$ (assuming that $\varepsilon \leq 1/2$). Since the clauses are perturbed independently, the probability that ψ satisfies F^* is at most

$$\left(1-\varepsilon^k\right)^m \leq e^{-m\cdot\varepsilon^k} < 2^{-n}.$$

Since there are 2^n ways to fix ψ, the proposition follows.

The calculation we just did assumes that every clause has k satisfied literals under ψ (when we upper bound the probability that ψ satisfies C), and therefore the clause "survives" with probability $1-\varepsilon^k$. To determine the correct asymptotic order of the dependence on ε, a more careful argument is needed, which takes into account the number of true literals in every clause. The following is an improved upper bound on $d_{\varepsilon,k}$.

Theorem 4. *(upper bound) Let F be any k-CNF instance with m clauses over n variables, and let $d = m/n$. Let ε be the perturbation parameter, and let F^* be the perturbed instance. There exists*

an absolute constant $c_1 > 0$ s.t. if

$$d \geq \frac{c_1}{\varepsilon^{k-1}} \cdot \log \frac{1}{\varepsilon}$$

then whp F^* is not satisfiable.

The theorem is proven in Section 2. After stating this result, the natural question is how tight this bound on d is? The following theorem establishes that the dominant term in the bound (that of ϵ^{1-k}) is tight.

Theorem 5. *(lower bound) There exists a constant c_2 and a k-CNF F with m clauses over n variables, and $d = m/n$ satisfying*

$$d \leq \frac{c_2}{\varepsilon^{k-1}} \cdot \frac{1}{k^2},$$

so that its ε-perturbation, F^*, is whp satisfiable. Furthermore $c_2 \geq 1/42$.

Remark 6. Our bounds are most informative when k is viewed as fixed, and $d_{\varepsilon,k}$ is viewed as an increasing function of $1/\varepsilon$. For some other settings of the parameters our results are weaker than known results. In particular, when $\varepsilon = 1/2$ the bound in Theorem 5 (namely, $2^k n/(84k^2)$) is weaker than the known lower bounds on the satisfiability threshold for random formulas in $\mathcal{F}_{n,m,k}$. Observe that a random formula in $\mathcal{F}_{n,m,k}$ can be viewed as a 1/2-smoothed random formula where all literals were initially positive.

1.2 A new upper bound on the running time of `Walksat`

The method we use to prove Theorem 5 has an appealing consequence. As part of the proof we analyze a certain random branching process on k-CNF formulas. As it turns out, this branching process is very useful to understand the behavior of the very popular `Walksat` algorithm on random $\mathcal{F}_{n,m,k}$ instances for $m/n = \Omega(2^k/k^2)$. Let us recall the `Walksat` algorithm suggested in [16] for example.

Let $V = \{x_1, \ldots, x_n\}$ be a set of n variables. Applied to a k-SAT formula F over V, `Walksat` proceeds as follows.

1. Pick a random assignment $\sigma \in \{0,1\}^V$. Insert all clauses of F that are not satisfied under σ into the FIFO queue Q in an arbitrary order.
2. While Q is non-empty, pop the first clause C from Q. If C is unsatisfied under σ, then pick a variable from C uniformly at random and flip its value in the assignment σ; if this yields any new unsatisfied clauses, insert these clauses into the end of the queue Q.
3. Output the assignment σ.

We have stated `Walksat` using the concept of a FIFO queue to store the currently violated clauses. There are other equally common implementations (e.g., one could choose the clause C in Step 2 uniformly at random from the set of currently violated clauses). Our analysis carries over to these other implementations easily.

`Walksat` is part of a broad family of local search algorithms. Algorithms in this family start with an assignment to the input formula, and gradually change it one bit at a time, by trying

to locally optimize a certain function. These algorithms (the most famous of which is `Walksat`) are close relatives of the simulated annealing method and were found to compete successfully with DLL-type algorithms.

Empirical results on random 3CNF formulas indicate that `Walksat` terminates successfully in linear time up to clause density 2.65 [17], which is approximately $2^k/k$ for $k = 3$. The best current rigorous analysis for $k = 3$ works up to clause-density 1.63 (which is the threshold for the pure literal rule to work *whp*) and is due to [3]. Indeed, the analysis of `Walksat` which is given in [3] relies on the pure literal rule to solve the formula. It was shown in [4] that the pure literal rule works *whp* for random k-CNF formulas in $\mathcal{F}_{n,m,k}$ only when $m/n \leq \omega_k$ for a certain strictly decreasing sequence ω_k that tends to 0 as k gets large. (For instance, $\omega_k < 1$ for $k = 9$.)

We are able to prove the following upper bound on the typical running time of `Walksat`.

Theorem 7. *Let F be a random instance from $\mathcal{F}_{n,m,k}$. If $k \geq 5$ and $m/n \leq 2^k/(30k^2)$, then on input F `Walksat` finds a satisfying assignment after flipping only $O(n/k)$ variables whp. Furthermore, if $k \geq 30$, then the same is true for $m/n \leq 2^k/(6k^2)$.*

Remark 8. For $k = 3$ it is shown in [3] that `Walksat` has a linear expected running time for $m/n < 1.63$ *whp*, and the same proof technique can be used to show that `Walksat` has a linear expected running time for $k = 4$ and $m/n < 1.54$. More generally, it is conceivable that the arguments from [3] can be used to show that for all $k \geq 3$ `Walksat` has a linear expected running time for densities m/n below the pure literal threshold. The density $2^k/(30k^2)$ exceeds the pure literal threshold for $k \geq 12$.

The conjectured threshold (supported by empirical evidence) up to which `Walksat` will work efficiently is $2^k/k$. Although our result doesn't verify this conjecture, we make a major step in establishing a ratio where `Walksat` works efficiently, which is not vanishing with k, and is just a factor $O(k)$ from the experimental observations. Another algorithm which is worth mentioning at this point is the Unit Clause Propagation (UCP) algorithm. Like `Walksat`, it too has a rather simple description and very natural guiding principles. The UCP algorithm was analyzed in [5]. Other variants of that algorithm were later analyzed in [6, 12]. The algorithms analyzed in those papers were rigorously shown to solve random k-CNF formulas with density up to $2^k/k$, and fail beyond that point. As for `Walksat`, $2^k/k$ remains a conjecture to be verified.

Another interesting point which may be mentioned in this context is the ratio $m/n = 2^k \log k/k$. The physicists call this point the dynamic 1RSB transition. At that point, the geometry of the solution space of a typical formula is believed to change from having a simple structure of one giant cluster of satisfying assignments, to a multi-clustered terrain. The existence of multi-clustered terrain for $m/n \geq (1+\alpha)2^k \log k/k$ was proven rigorously for $k \geq 8$ in [1] (α tends to 0 as k grows).

2 Proof of Theorem 4

The technique we use to prove the theorem is a careful first moment argument, which refines the argument used to prove Proposition 3.

We say that a formula F is w-expanding if for every $1 \leq t \leq n$, every subset of t variables participates in at least tw clauses.

Lemma 9. *Let F be a k-CNF instance with m clauses over n variables with density d (where $d = m/n$). F has a $d/2$-expanding subformula F'. Furthermore, F' contains at least $m/2$ clauses.*

Proof. As long as F contains a subset T of variables that participates in less than $|T|d/2$ clauses, remove these variables and all clauses that contain them. The resulting instance, F', satisfies the expansion condition of the lemma. As for the number of clauses in F', every set T of variables that was removed accounts for at most $|T|d/2$ clauses that were removed from F. Let $T_1, T_2, \ldots T_p$ be the sets that were removed in the iterative procedure (and let $t_i = |T_i|$), then

$$\begin{aligned} |F'| &\geq m - \sum_{i=1}^{p} t_i d/2 \\ &\geq m - d/2(\sum_{i=1}^{p} t_i) \geq m - dn/2 = m/2. \end{aligned}$$

■

In our proof of Theorem 4 we will consider a $d/2$-expanding subformula F', and prove that its ε-smoothed version F'^* is unlikely to be satisfiable. This of course implies that also the ε-smoothed version F^* of F is unlikely to be satisfiable, because F'^* is a subformula of F^*. Formally, we should denote the number of variables in F' by n', the number of clauses by m' (noting that $m' \geq \max[n'd/2, m/2]$) and its density by $d' \geq d/2$. However, to simplify notation we shall rename F' by F, n' by n, m' by m and $d/2$ by d (resulting in F being d-expanding). Note that now d is a lower bound on the density rather than being equal to m/n.

Given a k-CNF formula F and some assignment ψ to its variables we denote by $\psi_i(F)$ the number of clauses in F which have exactly i true literals under ψ (for $i = 0, \ldots, k$). Using this notation, the probability that an assignment ψ survives the ε-perturbation is exactly

$$\prod_{i=0}^{k} \left(1 - \varepsilon^i (1-\varepsilon)^{k-i}\right)^{\psi_i(F)}. \tag{2.1}$$

In the proof of Proposition 3 we assumed the "worst case", namely that $\psi_k(F) = |F|$ for every assignment ψ, and took the union bound over all 2^n possible assignments. Here we shall distinguish in (2.1) between clauses that contribute to $\psi_k(F)$ and and clauses that contribute to $\psi_{k'}(F)$ for $k' < k$. Thus we shall bound:

$$\prod_{i=0}^{k} \left(1 - \varepsilon^i (1-\varepsilon)^{k-i}\right)^{\psi_i(F)} \leq \left(1 - \varepsilon^k\right)^{\psi_k(F)} \cdot \left(1 - (1-\varepsilon)\varepsilon^{k-1}\right)^{|F| - \psi_k(F)}. \tag{2.2}$$

Equation (2.2) will be incorporated in a first moment enumeration of all possible assignments. The key point in this enumeration, formally stated in the following lemma, is to observe that there aren't too many assignments for which $\psi_k(F)$ is too large.

Lemma 10. *Let F be a d-expanding k-CNF formula on n variables and m clauses. There are at most $n\binom{n}{g/d}$ assignments satisfying $\psi_k(F) = m - g$.*

Proof. Consider any two assignments ψ, φ satisfying $\psi_k(F) = \varphi_k(F) = m - g$. Let T be the set of variables on which they disagree and denote its cardinality (which corresponds to the Hamming

distance between the two assignments) by t. Observe that all clauses in which a variable from T appears cannot contribute to both $\psi_k(F)$ and $\varphi_k(F)$. There are at least td such clauses (by expansion), and at least half of them didn't contribute to, say, $\psi_k(F)$. Therefore $g \geq td/2$. Rearranging, $t \leq 2g/d$. Using Kleitman's theorem for the volume of a diameter-t subset of the n-dimensional binary cube [13], the total number of assignments ψ such that $\psi_k(F) = m - g$ is at most

$$\sum_{i=0}^{t/2} \binom{n}{i} \leq n \binom{n}{g/d}.$$

The inequality holds since $t/2 \leq n/2$, and therefore the maximum is obtained at $t/2$. ■

We are now ready to prove Theorem 4. Recall (2.2) – the upper bound on the probability that an assignment ψ satisfies F after the ε-perturbation, and observe that $1 - \varepsilon \geq 1/2$. Parameterizing by the number of clauses not satisfied k times and using the above lemma we obtain the following union bound on the probability that F^* is satisfiable:

$$n \sum_{g=0}^{m} \binom{n}{g/d} \left(1 - \varepsilon^k\right)^{m-g} \left(1 - \frac{\varepsilon^{k-1}}{2}\right)^g. \tag{2.3}$$

We break the above sum into two sums, one for values of g up to εdn and the other for values of g above εdn. Bounding the first of these sums by its largest term and using the fact that $m \geq dn$ we obtain:

$$n \sum_{g=0}^{\varepsilon dn} \binom{n}{g/d} \left(1 - \varepsilon^k\right)^m \leq \varepsilon dn^2 \binom{n}{\varepsilon n} \left(1 - \varepsilon^k\right)^{dn} \tag{2.4}$$

Using $\binom{n}{\varepsilon n} \simeq e^{\varepsilon n(1 - \ln \varepsilon)}$, the term $\binom{n}{\varepsilon n} \cdot \left(1 - \varepsilon^k\right)^{dn}$ is smaller than 1 (and decreases exponentially with n) when $d > \frac{5}{\varepsilon^{k-1}} \ln(\frac{1}{\varepsilon})$ (here the choice of constant 5 is for concreteness rather than an optimal choice). The term εdn^2 is then cancelled out if in addition we have that $\varepsilon > c/n$ for some sufficiently large constant c (essentially we need $c > k$).

For the second of these sums we upper bound it by discarding the term $(1 - \varepsilon^k)^{m-g}$ and identifying the value of g that contributes most to the sum

$$\sum_{g=\varepsilon dn}^{m} \binom{n}{g/d} \left(1 - \frac{\varepsilon^{k-1}}{2}\right)^g$$

Observe that $\binom{n}{g/d}$ ranges from roughly $e^{\varepsilon n(1 - \ln \varepsilon)}$ (when g is at its lowest value εdn) to at most 2^n (when g is around $\frac{1}{2}dn$). It is convenient to view g as $\varepsilon' dn$ for $\varepsilon \leq \varepsilon' \leq 1/2$ and then $\binom{n}{g/d} \simeq e^{\varepsilon n(1 - \ln \varepsilon')}$. For our value of $d > \frac{5}{\varepsilon^{k-1}} \ln(\frac{1}{\varepsilon})$ the term $(1 - \frac{\varepsilon^{k-1}}{2})^g$ then becomes at most $e^{-\frac{5}{2}\varepsilon' n \ln(\frac{1}{\varepsilon})}$ and the product of the two terms is smaller than 1 (note that $\frac{1}{\varepsilon} \geq \frac{1}{\varepsilon'}$) and decreases exponentially with n. Since the summation involves essentially at most $dn/2$ terms (for larger values of g the terms decrease geometrically) we conclude that for our value of d and for $\varepsilon > c/n$, the second sum is also negligible. This concludes the proof of Theorem 4.

3 Proof of Theorem 5

In this section we will show that there exists a k-CNF formula on n variables with dn clauses, $d \leq \varepsilon^{1-k}/(42k^2)$, such that the ε-perturbed instance F^* is *whp* satisfiable.

Let F be the formula generated via the following random procedure:

Procedure 11. *Go over all $\binom{n}{k}$ clauses that contain only positive literals; include each such clause with probability $p = dn/\binom{n}{k}$.*

Indeed *whp* F has basically dn clauses (just apply the Chernoff bound). It remains to show that F^* is *whp* satisfiable. In what follows we shall prove that for all but $o(1)$-fraction of the formulas F, F^* is *whp* satisfiable. Thus we prove a much stronger statement – we show that there exist many formulas of size dn whose ε-perturbation is *whp* satisfiable.

To prove the statement we show that the all-TRUE assignment, which is clearly satisfying for F, can be typically adjusted to a satisfying assignment of F^*. Actually, what we will show is that this adjustment can be found efficiently, for example – using the pure literal rule (or, the `Walksat` algorithm). We divide the variables into two sets. One is the set of variables that stay TRUE in the satisfying assignment of F^* that we derive (if indeed F^* is satisfiable), and the others are variables that may be changed to FALSE. We call the latter *feeble* variables. The set of feeble variables is defined via the following recursive procedure:

1. Set $A_0 = \{x | \exists(\bar{x} \vee \bar{y}_1 \vee \ldots \vee \bar{y}_{k-1})$ in $F^*\}$.
2. Set $A_i = \{x |$ $\exists$ clause containing $\bar{x}$ in which all non-negated variables belong to $\bigcup_{j \leq i-1} A_j\}$.
3. Set $A = \bigcup_{i \geq 0} A_i$, and call it the set of feeble variables.

In the definition of A_i we don't consider variables that already belong to some A_j, $j < i$. Given a set L of variables, we denote by $F[L]$ the subformula of F that contains all clauses where all k literals belong to L. We also call it the subformula induced by L.

Remark 12. The set A plays an important role in the analysis of the `Walksat` algorithm (to be carried in Section 4). Consider any k-CNF F (not necessarily satisfiable), and $A = A(F)$ defined as above. Let E be a particular execution (maybe infinite) of the `Walksat` algorithm on F when starting from the all-TRUE assignment, and let T_E be the set of variables flipped in that execution. One can easily verify that if we let $T = \cup_E T_E$, then always $T \subseteq A$. Therefore (loosely speaking) proving that `Walksat` is typically efficient reduces to asserting that A typically has a simple structure. (Of course there is the issue of the starting point, `Walksat` doesn't "know" to start at the all-TRUE assignment. This point is discussed in Remark 19).

Proposition 13. *Let F be generated via the random process defined in Procedure 11 above, and let F^* be its ε-perturbation. Every assignment that sets all variables in $V \setminus A$ to TRUE satisfies all clauses in $F^* \setminus F^*[A]$.*

Proof. By contradiction, assume that there exists an assignment which sets all variables in $V \setminus A$ to TRUE but there exists a clause in $F^* \setminus F^*[A]$ which is not satisfied. Consider the variables in that clause which do not belong to A (there must be at least one by the choice of the clause). It must be that all of them are negated, otherwise the clause is satisfied. But then, according to our

definition of A_i, all these variables will be part of A by definition, deriving a contradiction. ■

It remains to show the following:

Proposition 14. *Let F be generated via the random process defined in Procedure 11 above, and let F^* be its ε-perturbation. If $d \leq \varepsilon^{1-k}/(42k^2)$, then whp $F^*[A]$ is satisfiable (the probability is taken over the choice of F and F^*).*

To prove that, we show that typically the pure literal rule solves $F^*[A]$, and thus in particular $F^*[A]$ is satisfiable. The pure literal rule is the following simple procedure:

> *Repeat while possible: pick a variable that appears only in one polarity, set it in a satisfying manner, and remove all clauses in which it appears.*

Remark 15. The reader may recall us stating earlier that the pure literal rule fails to solve formulas with high density like F^* [4]. Note that we are not claiming that the pure literal procedure terminates successfully when applied to F^*, but rather to a carefully chosen subformula of F^*, $F^*[A]$.

Let us start by establishing a structural property which implies the success of the pure literal rule.

Definition 16. *A variable x is called* pure *in F if it appears only in one polarity. A formula F is called* complicated *if all its variables are not pure. A formula is called* degenerate *if none of its subformulas is complicated.*

Clearly, if F is degenerate, then the pure literal method solves it.

Proposition 17. *Let F be generated via the random process defined in Procedure 11 above, and let F^* be its ε-perturbation. If $d \leq \varepsilon^{1-k}/(42k^2)$ then whp (over the choice of F and F^*), every subformula F' of F^* induced by at most $\varepsilon n/k$ variables is degenerate.*

Proof. If F' is not degenerate then it contains a subformula F'' on a set of variables S of size $s = |S| \leq \varepsilon n/k$ so that every variable in S appears at least twice, that is $|F''| \geq \mu \geq 2s/k$, and at least once negatively.

Suppose that $\binom{n}{k}p = m = \zeta\varepsilon^{1-k}n/k^2$ for $\zeta \leq 1/42$. Let S be a set of variables of size $s \leq \varepsilon n/k$. Set $t = \mu/(2s)$, if F'' has no pure literal then $t \geq 1/k$. Moreover, if F'' has no pure literal, then each variable in S occurs negatively at least once. This implies that among the $k\mu$ literal occurrences at least s are negative. The probability of this event is at most

$$Q_{s,t} = \binom{k\mu}{s}\varepsilon^s.$$

Moreover, the probability that S spans at least μ clauses is at most

$$P_s = \binom{\binom{s}{k}}{\mu}p^\mu.$$

Setting $p = m/\binom{n}{k}$, $m = \zeta\varepsilon^{1-k}n/k^2$, and using $\mu \geq 2s/k$ we obtain:

$$P_s \leq \left(e\left(\frac{s}{n}\right)^{k-1}\frac{km}{n}\right)^\mu.$$

Finally, there are

$$H_s = \binom{n}{s} \leq \left(\frac{en}{s}\right)^s$$

ways to choose the set S. Hence, the probability that *there is* a set S of size s with μ clauses and without a pure literal is at most

$$\begin{aligned} H_s P_s Q_s &\leq \left(\frac{en}{s}\right)^s \left(\frac{ek\mu\varepsilon}{s}\right)^s \left(e\left(\frac{s}{n}\right)^{k-1}\frac{km}{n}\right)^{2st} \\ &\leq \left[e\left(\frac{2e^2knt\varepsilon}{s}\right)^{1/(2t)}\left(\frac{s}{n}\right)^{k-1}\frac{km}{n}\right]^{\mu} \\ &\leq \left[\zeta e\left(\frac{2e^2knt\varepsilon}{s}\right)^{1/(2t)}\left(\frac{s}{n}\right)^{k-1}\frac{\varepsilon^{1-k}}{k}\right]^{\mu} \equiv y^{\mu}. \end{aligned}$$

Our next goal is to show that this quantity is $o(1)$. To this end we analyze y. The term $(2e^2t)^{1/(2t)} < 15$ for every t. Therefore $\zeta e(2e^2t)^{1/(2t)} < 1$ for $\zeta \leq 1/42$. Hence, it suffices to bound

$$z = \frac{k^{1/(2t)-1}}{2}\cdot\left(\frac{\varepsilon n}{s}\right)^{1/(2t)}\left(\frac{s}{n}\right)^{k-1}\varepsilon^{1-k} = \frac{k^{1/(2t)-1}}{2}\cdot\left(\frac{s}{\varepsilon n}\right)^{k-1-1/(2t)}.$$

Using our assumption that $s/n < \varepsilon/k$, the latter is at most $k^{-k+1/t}/2 \leq 1/2$ as $t \geq 1/k$.

To conclude the proof we sum over all possible sizes $s \leq \varepsilon n/k$. If $s = o(n)$ then $z^{\mu} \leq o(n^{-1})$; if $s = \Omega(n)$, but still $s \leq \varepsilon n/k$, z^{μ} is exponentially small in n, and in particular $z^{\mu} \leq o(n^{-1})$. Finally, observe that s has at most n possible values. ■

To conclude the proof of Proposition 14 it suffices to show that *whp* $|A| \leq \varepsilon n/k$.

Proposition 18. *Let F be generated via the random process defined in Procedure 11 above, and let F^* be its ε-perturbation. If $d = \zeta\varepsilon^{1-k}/(3k^2)$, $\zeta \leq 1$ then whp (over the choice of F and F^*) $|A| \leq \zeta\varepsilon n/k$.*

Proof. It will be convenient for the sake of the proof to view the process by which F, F^* and A are generated as if the decision of which variables appear in each clause are deferred until the point when they actually influence the construction of A.

Let the number of clauses in F be $m = dn$. This defines for us km locations in the formula that will need to be filled in randomly by variables. Before deciding on the variables, the polarity of every location is set independently at random to be positive with probability $1-\varepsilon$ and negative with probability ε. Now, we start constructing A. In this process, for the variables that enter A we shall reveal their locations in F. We shall also maintain a list B of clauses that contribute towards the construction of A. A clause may enter B only if it is *dangerous* in the following sense: every one of its k locations is either negated, or contains a variable that is already in A.

Given the above, A is constructed (and parts of F^* are revealed) by repeatedly applying the following procedure.

1. Pick the first clause C that is dangerous but not yet in B. If there is no such clause, stop.

2. For every location of C that is still not revealed (these are the variables not currently in A, and are necessarily negated), assign to it a variable at random from the set of variables that are still not in A. Add these variables to A in the order of their appearance in C, and add the clause C to B.

3. For every location in F that does not yet contain a variable, decide independently at random (with the natural marginal probabilities) whether to put there one of the new variables that entered A through B. If yes, then rather than writing the name of the variable in this location, write its index with respect A, where variables are indexed in the order in which they enter A. (Namely, write i if the variable is the ith variable to ever enter A.)

When the above process stops, then to complete the construction of F^*, assign variables at random to the locations that have not yet been revealed using the natural marginal distribution (choosing them only from variables that are not in A, and without choosing the same variable more than once in the same clause).

The distribution over formulas F^* generated by the above process is essentially identical to that of picking every positive clause independently with probability p, and then flipping each literal with probability ε (Procedure 11 and the smoothing operation). (Technically, there are two minor differences. One is that in our process we fix the number of clauses to be exactly m, rather than some distribution concentrated around m. The other is that our process might generate two clauses with the same set of variables, though the probability of this happening is small. Both differences have virtually no effect on the analysis, and we omit the standard details of how they can be formally handled.) Note that our process uses two different names for some of the variables: their original name in some locations, and their index with respect to A in others. However, there is no ambiguity regarding the correspondence between the naming systems, because given F^* (with this naming system), our process for generating A is deterministic and the order in which variables enter A is well defined.

A crucial observation regarding all clauses in B is that each of their locations is either negated, or contains an index (rather than a name of a variable), and this index lies between 1 and $|A|$. Moreover, for every positive location in F (regardless of whether it is in B or not) and any index $i \leq |A|$, the probability that the location contains the index i is exactly $1/n$.

Now we wish to select a value of $0 < \delta < 1$ such that with overwhelming probability, $|A| \leq \delta n$. Note that A increases only by dangerous clauses, and so for $|A|$ to ever reach δn, it must be the case that $|B|$ reached at least $\delta n/k$. Moreover, for every clause of B up to that point, it must be the case that all locations are either negated or contain an index of value at most δn. Now a simple computation shows that with high probability, δ cannot be too large.

Consider the process by which A expands at the time when $|A| = \delta n$. The probability that a clause is dangerous (and hence may potentially take part in the process) is at most $(\varepsilon + \delta)^k$. Let I'_j be an indicator random variable which is 1 with probability $(\varepsilon + \delta)^k$, and let $B' = \sum_{j=1}^{m} I'_j$. B is statistically dominated by B'. To see this write B as the sum of indicator variables I_j, where I_j is 1 if the j'th clause in F that we are revealing is dangerous (given clauses $1, \ldots, j-1$). Although we do not show an explicit expression for $E[I_j]$, we argued before that $E[I_j] \leq (\varepsilon + \delta)^k$. Since B' *whp* does not exceed $1.01dn(\varepsilon + \delta)^k$, this is also the case for B.

To conclude, *whp* $|B| \leq 1.01dn(\varepsilon+\delta)^k$, and also (always) $|B| \geq |A|/k$. Thus assuming $|A| \geq \delta n$ implies the inequality $1.01dn(\varepsilon + \delta)^k \geq \delta n/k$. However, choosing $d = \frac{\zeta\varepsilon^{1-k}}{3k^2}$ and $\delta = \zeta\varepsilon/k$, the

required inequality fails to hold, and hence *whp* $|A| < \zeta\varepsilon n/k$. ■

4 Proof of Theorem 7

In this section we assume that $k \geq 5$. Recall the definition of $\mathcal{F}_{n,m,k}$ (referred to in Theorem 7): fix $c, n > 0$ (c may depend on n), choose $m = cn$ clauses uniformly at random out of $M = 2^k\binom{n}{k}$ possible ones. The analysis we had in Section 3 was more similar in flavor to the following variation of $\mathcal{F}_{n,m,k}$: go over the M possible clauses and include each one with probability $p = p(n,k)$. Let $\mathcal{F}_{n,p,k}$ be the last model. Standard calculation show that when $p = m/M$, every property that holds with probability $1-q$ in $\mathcal{F}_{n,m,k}$ holds with probability $1 - O(q\sqrt{m})$ in $\mathcal{F}_{n,p,k}$. In this section we will analyze $\mathcal{F}_{n,p,k}$, showing that all sufficient conditions to allow *whp* a linear running time hold with probability at least $1 - o(m^{-1/2})$, and therefore in particular hold *whp* in $\mathcal{F}_{n,m,k}$.

In order to use the analysis carried in the previous section, observe that if generating F according to Procedure 11, and perturbing it with $\varepsilon = 1/2$, then F is distributed like $\mathcal{F}_{n,p,k}$. (This is true up to the following issue. According to the way we generated F there cannot be two clauses that have the same set of variables, whereas had F been sampled according to $\mathcal{F}_{n,p,k}$ that would have been possible (in that case F may contain, say, $(\bar{x}_1 \vee x_{17} \vee \bar{x}_{23})$ and $(\bar{x}_1 \vee \bar{x}_{17} \vee x_{23})$). However, the probability of such a pair of clauses ever appearing in $F \in \mathcal{F}_{n,p,k}$ is $O(m^2 \cdot n^{-k})$ which is $o(n^{-1})$ for $p = m/M$, $m = O(n)$, and $k \geq 3$. Therefore the two models are statistically sufficiently close so that we can also switch back to $\mathcal{F}_{n,m,k}$.)

Let A' be the set of variables defined as follows: $x \in A'$ iff there exists an execution of `Walksat`, starting from the all-TRUE assignment, such that x is flipped by `Walksat`. It is easy to see that $A' \subseteq A$ (where A is the set of feeble variables defined in the previous section).

Remark 19. Although we assume that `Walksat` starts from the all-TRUE assignment, it is done just for convenience. Suppose that in Procedure 11 we don't use the all-TRUE assignment, but some arbitrary assignment ψ, and include only clauses where all k literals are true under ψ. For $\varepsilon = 1/2$, the distribution of the perturbed instance F^* remains the same. Therefore by symmetry, it does not matter from what assignment `Walksat` starts as we can always think of it as the assignment that was used to generate F.

In comparison to [3], the new insight here is that the set of variables flipped by `Walksat` is confined to A (the feeble variables) i.e., it suffices to analyze the performance of `Walksat` on $F[A]$. Since Proposition 18 shows that A is "small" *whp*, the formula $F[A]$ has a rather simple structure. More precisely,

Lemma 20. *Let F be a random formula generated via Procedure 11, $d = \zeta 2^k/k^2$ for some suitably chosen constant $\zeta = \zeta(k)$ (to be determined in the proof). Let A be the set of feeble variables defined in Section 3. Then F enjoys the following property with probability $1 - o(n^{-1})$: for any $S \subset V$ of size $|S| \leq |A|$ the formula $F[S]$ contains at most $\frac{3|S|}{2k}$ clauses.*

Proof. We use a first moment argument. For a given size $s = \beta n$ there are $\binom{n}{s}$ ways to choose the set S. Moreover, letting $\rho = 3/(2k)$, the probability that any given S of size s contains $t = \rho s$

clauses is

$$\binom{\binom{s}{k}}{t} p^t \leq \left(\frac{e\binom{s}{k}dn}{t\binom{n}{k}}\right)^t \leq \left(\frac{eds^{k-1}}{\rho n^{k-1}}\right)^t \leq \left(\frac{e2^{k+1}\zeta\beta^{k-1}}{3k}\right)^{\rho s}.$$

Hence, the first moment is at most

$$\binom{n}{s}\left(\frac{e\zeta 2^k s^{k-1}}{6\rho k^2 n^{k-1}}\right)^{\rho s} \leq \left[\frac{e}{\beta}\cdot\left(\frac{e2^{k+1}\zeta\beta^{k-1}}{3k}\right)^{\rho}\right]^s = \left[e\left(\frac{e2^{k+1}\zeta}{3k}\right)^{\rho}\beta^{(k-1)\rho-1}\right]^s.$$

Observe that the exponent of β is positive for all $k \geq 5$. Hence, if β gets sufficiently small the entire expression is $o(1)$. Furthermore, Proposition 18 shows that $\beta \leq 3\zeta/k$ *whp* if $\zeta \leq 1/6$. Hence, if either ζ gets sufficiently small or k gets sufficiently large, the first moment will be $o(1)$. A detailed calculation shows that $\zeta = 1/30$ suffices for all $k \geq 5$, and $\zeta = 1/6$ is small enough if $k \geq 30$. ■

Lemma 20 holds for F^* (which is F perturbed with $\varepsilon = 1/2$) with the same probability (as we just flip polarities). As we mentioned above, F^* is statistically close to a $\mathcal{F}_{n,p,k}$. As the probability of Lemma 20 is sufficiently large, it is true *whp* for an $\mathcal{F}_{n,m,k}$ instance (for $m = dn$) as well.

To conclude the proof of the theorem, consider the following bipartite graph where one part is the variables of A and the other the clauses in $F^*[A]$. Lemma 20 implies that this graph contains a $\lfloor 2k/3 \rfloor$-fold matching M from the set of clauses to the set A. That is, M connects each clause of $F^*[A]$ with exactly $\lfloor 2k/3 \rfloor$ variables, and each variable occurs in at most one edge of M.

Finally, observe that the existence of such a matching implies that $F^*[A]$ has a satisfying assignment φ where in every clause at least $\lfloor 2k/3 \rfloor$ of the literals are satisfied. To see this, define an assignment σ as follows. For all variables $v \in A$ that are incident with an edge $\{C, v\}$ of M, let $\sigma(v)$ =TRUE if v occurs positively in C, and $\sigma(v)$ =FALSE if v occurs in C negatively. The variables $v \in A$ that are not incident with an edge of M can be assigned arbitrarily. Since M connects each clause with at least $\lfloor 2k/3 \rfloor > k/2$ variables, σ satisfies more than $k/2$ literals in each clause of $F^*[A]$. This implies that each time `Walksat` flips a variable, the probability that the Hamming distance between `Walksat`'s current assignment and the assignment σ decreases is strictly bigger than $\frac{1}{2}$. Hence, the theory of random walks guarantees that *whp* `Walksat` reaches a satisfying assignment after performing at most $O(|A|) = O(n/k)$ flips *whp* (as *whp* $|A| \leq O(n/k)$). This completes the proof of Theorem 7.

References

[1] D. Achlioptas and A. Coja-Oghlan. Algorithmic barriers from phase transitions. *preprint*, 2008.

[2] D. Achlioptas and Y. Peres. The threshold for random k-SAT is $2^k \log 2 - O(k)$. *Journal of the AMS*, 17(4):947–973, 2004.

[3] M. Alekhnovich and E. Ben-Sasson. Linear upper bounds for random walk on small density random 3-cnfs. *SIAM J. on Comput.*, 36(5):1248–1263, 2007.

[4] A. Z. Broder, A. M. Frieze, and E. Upfal. On the satisfiability and maximum satisfiability of random 3-CNF formulas. In *Proc. 4th ACM-SIAM Symp. on Discrete Algorithms*, pages 322–330, 1993.

[5] M. Chao and J. Franco. Probabilistic analysis of a generalization of the unit clause selection heuristic for the k-satisfiability problem. *Information Sciences*, 51:289–314, 1990.

[6] V. Chvátal and B. Reed. Mick gets some (the odds are on his side). In *Proc. 33rd IEEE Symp. on Found. of Comp. Science*, pages 620–627, 1992.

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

[8] U. Feige. Refuting smoothed 3CNF formulas. In *Proc. 48th IEEE Symp. on Found. of Comp. Science*, 2007.

[9] U. Feige and E. Ofek. Easily refutable subformulas of large random 3cnf formulas. *Theory of Computing*, 3(1):25–43, 2007.

[10] A. Flaxman. A spectral technique for random satisfiable 3CNF formulas. In *Proc. 14th ACM-SIAM Symp. on Discrete Algorithms*, pages 357–363, 2003.

[11] E. Friedgut. Sharp thresholds of graph properties, and the k-sat problem. *J. Amer. Math. Soc.*, 12(4):1017–1054, 1999.

[12] A. Frieze and S. Suen. Analysis of two simple heuristics on a random instance of k-SAT. *J. of Algorithms*, 20:312–355, 1996.

[13] D. Kleitman. On a combinatorial conjecture of Erdős. *J. Combinatorial Theory*, pages 209–214, 1966.

[14] M. Krivelevich, B. Sudakov, and P. Tetali. On smoothed analysis in dense graphs and formulas. *Random Structures and Algorithms*, 29(2):180–193, 2006.

[15] L. Levin. Average case complete problems. *SIAM J. on Comput.*, 15(1):285–286, 1986.

[16] C. Papadimitriou. On selecting a satisfying truth assignment. In *Proc. 32nd IEEE Symp. on Found. of Comp. Science*, 1991.

[17] A. Parke. Scaling properties of pure random walk on random 3SAT. In *Proceedings of the 8th International Conference on Principles and Practice of Constraint Programming*, 2002.

[18] D. Spielman and S. Teng. Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. *J. of the ACM*, 51(3):385–463, 2004.