On the Complexity of Unsatisfiability Proofs for Random k-CNF Formulas

Paul Beame* Richard Karp† Toniann Pitassi‡ Michael Saks§

Abstract

We study the complexity of proving unsatisfiability for random k-CNF formulas with clause density $\Delta = m/n$ where m is number of clauses and n is the number of variables. We prove the first nontrivial general upper bound, giving algorithms that, in particular, for $k = 3$ produce refutations almost certainly in time $2^{O(n/\Delta)}$. This is polynomial when $m \geq n^2/\log n$.

We show that our upper bounds are tight for certain natural classes of Davis-Putnam algorithms. We show further that random 3-CNF formulas of clause density Δ almost certainly have no resolution refutation of size smaller than $2^{\Omega(n/\Delta^{4+\varepsilon})}$, which implies the same lower bound on any Davis-Putnam algorithm. We also give a much simpler argument based on a novel form of self-reduction that yields a slightly weaker $2^{\Omega(n/\Delta^{5+\varepsilon})}$ lower bound.

1 Introduction

The random k-CNF model has been widely studied for several good reasons. First, it is an intrinsically natural model, analogous to the random graph model, that sheds light on fundamental structural properties of the satisfiability problem. Second, for appropriate choice of parameters, randomly chosen formulas are empirically difficult for satisfiability, and are a commonly used benchmark for testing satisfiability algorithms. (See, for example, the encyclopedic survey of the SAT problem in [GPFW97].) Lastly, the random model is important for proving lower bounds for propositional proof systems. Lower bounds for random k-CNF formulas attest to the fact that the proof system in question is ineffective on average.

A fundamental conjecture about the random k-CNF formula model, (see [CS88, BFU93, CF90, CR92, FS96, KKK96]) says that there is a constant c_k, the *satisfiability threshold*, such that a random k-CNF formula of clause density Δ is almost certainly satisfiable for $\Delta < c_k$ (as n gets large), and almost certainly unsatisfiable if if $\Delta > c_k$. There is considerable empirical and analytic evidence for this. Recently Friedgut [Fri] showed that for each n there is a threshold $c_k(n)$ with the above property, but he does not rule out the possibility that $c_k(n)$ varies with n. It is known that $c_2 = 1$ is independent of n [CR92, Goe96], and that for each k $c_k(n)$ is bounded between two constants b_k and d_k that are independent of n, e.g., and $3.003 \leq c_3(n) \leq 4.598$ [FS96, KKK96].

The threshold indicates three distinct ranges of clause density for investigating complexity. For Δ at the threshold, an effective algorithm must be able to distinguish between unsatisfiable and satisfiable instances. Below the threshold, a random formula is almost certainly satisfiable and the problem of interest is to find a satisfying assignment quickly. Above the threshold, the formula is almost certainly unsatisfiable and we have the two closely related questions (i) what is the typical size of the smallest unsatisfiability proof? and (ii) how quickly can an algorithm find a proof?

Our attention here is restricted to the class of algorithms commonly known as *Davis-Putnam procedures*, and on unsatisfiability proofs using general *resolution*. A Davis-Putnam procedure, run on an unsatisfiable formula, produces a resolution proof for F. Therefore, the minimum size $\mathrm{res}(F)$ of a resolution proof for F, is a lower bound on the running time of any Davis-Putnam procedure on F, and, conversely, an upper bound on the running time of some Davis-Putnam algorithm on F provides an upper bound on $\mathrm{res}(F)$.

*Research supported by NSF grant CCR-9303017. Computer Science and Engineering, University of Washington, Box 352350, Seattle, WA 98195, `beame@cs.washington.edu`

†Computer Science and Engineering, University of Washington Box 352350, Seattle, WA 98195, `karp@cs.washington.edu`

‡ Research supported by NSF grant CCR-9457782 and US-Israel BSF Grant 95-00238. Computer Science Department, University of Arizona, Tucson, AZ 85721, `toni@cs.arizona.edu`

§ Research supported by NSF grant CCR-9700239. Department of Mathematics, Rutgers University, New Brunswick, NJ, `saks@math.rutgers.edu`. This work was done while on sabbatical at University of Washington.

Several empirical studies of Davis-Putnam procedures on random 3-CNF formulas have been done, e.g., by Selman, Mitchell, and Levesque [SML96] and Crawford and Auton [CA96]. The former applies a simple Davis-Putnam procedure to random 3-CNF formulas for various values of the clause density Δ. The curves in [SML96, CA96] show very low complexity for Δ below the threshold, a precipitous increase in complexity at the threshold, and a speedy decline to low complexity above the threshold.

Much has been made of the analogy with statistical physics [KS94], and there has been a suggestion that the computational complexity at the threshold is evidence of a critical phenomenon in complex systems and based on underlying chaos present only near the threshold. The empirical observation that satisfiability is easy below threshold is supported by analytical work. The proofs of the aforementioned lower bounds on c_k were obtained by showing that an appropriate version of the Davis-Putnam algorithm almost certainly finds a satisfying assignment in linear time, provided that Δ is below some specified constant.

The case of Δ above threshold is less well understood and is the focus of the present paper. In this case, there seem to be no previously known non-trivial upper bounds on the running time of algorithms on random instances. There are lower bounds, which are provided by lower bounds on $\mathrm{res}(F)$. In a seminal paper which provides much of the inspiration for our work, Chvátal and Szemerédi [CS88] showed that for any fixed Δ above the threshold there is a constant $\kappa_\Delta > 0$ such that $\mathrm{res}(F) \geq 2^{\kappa_\Delta n}$ almost certainly if F is a random k-CNF formula of clause density Δ. On the other hand, Fu [Fu95] showed that $\mathrm{res}(F)$ is almost certainly polynomial in n for $\Delta = \Omega(n^{k-2})$. These results, and the empirical work discussed above, raise the question of understanding the typical behavior of $\mathrm{res}(F)$ for random formulas F as a function of Δ. The lower bound in [CS88] as presented does not give bounds on the dependence of κ_Δ on Δ, but rough estimates show that for 3-CNF formulas the bound decreases as $1/\Delta^{\Omega(\Delta^4)}$. This implies that the lower bound declines extremely quickly and becomes trivial at very small non-constant clause densities, even density $O(\log^{1/4} n)$, leaving a huge gap between the upper and lower bounds. Fu [Fu95] extended the lower bounds to higher Δ in the case $k \geq 5$, and Beame and Pitassi [BP96] improved the bounds further. In particular, for random 3-CNF they gave an almost certain $\Omega(2^{\sqrt{n}/\Delta^{5/2+\varepsilon}})$ lower bound on $\mathrm{res}(F)$, which gives a non-trivial lower bound for clause density $\Delta = O(n^{1/(5+2\varepsilon)})$. The transition between the bounds of [CS88] and these bounds as Δ increases is very abrupt, leaving open the possibility that $\mathrm{res}(F)$ drops off precipitously as Δ increases beyond the threshold.

In this paper, we give the first asymptotic analysis of $\mathrm{res}(F)$ for random k-CNF formulas as a function of Δ for Δ above the threshold. We prove upper and lower bound results showing that κ_Δ decays as a fixed power of Δ, showing that there is not an isolated point of complexity at the threshold, but rather a slow and gradual decline in complexity as Δ increases.

In section 3, we give the first non-trivial general upper bound for random k-CNF formulas by showing that a very simple Davis-Putnam-like procedure achieves a $2^{O(n/\Delta^{1/(k-2)})}$ running time almost certainly, which implies the same upper bound on $\mathrm{res}(F)$. In particular, this is polynomial when $\Delta = \Omega((n/\log n)^{k-2})$, which is a slight improvement on Fu's result. The basic idea we use is to leverage the unsatisfiability properties of the well-understood 2-CNF formulas. We extend the analysis to show that similar upper bounds hold for the extremely simple algorithm used in [SML96].

In section 4, we prove lower bounds on the size of Davis-Putnam proofs for 3-CNF formulas. For the particular algorithm used by [SML96] we essentially match the upper bound of section 3 for all values of Δ. For large Δ, of size $\Omega(n^{3/2} \log^{\varepsilon} n)$, we prove, more generally, that a similar lower bound holds for any Davis-Putnam procedure whose splitting rule is independent of the particular formula. A key part of the analysis is to bound the number of unit clauses that can be generated in any Davis-Putnam procedure.

In section 5, we prove more general, but weaker, lower bounds on $\mathrm{res}(F)$ for random k-CNF formulas; these lower bounds thus apply to the running time of any Davis-Putnam procedure. We show that $\mathrm{res}(F) = 2^{\Omega(n/\Delta^{4+\varepsilon})}$ almost certainly for a random 3-CNF formula with clause density Δ. This uniformly subsumes and improves all previous lower bounds for such formulas. We first give a simple argument of a slightly weaker bound of $2^{\Omega(n/\Delta^{5+\varepsilon})}$ via a novel form of random self-reduction. This proof follows the structure of the the lower bound proof from [BP96], but replaces random restrictions by a technique called random augmentation, where we add randomly chosen clauses to the formula. This technique may be of independent interest. The stronger bound of $2^{\Omega(n/\Delta^{4+\varepsilon})}$ is obtained by extending [BP96] in another direction, and is considerably more complicated. As this article was going to press, we learned of a recent result of Ben Sasson and Wigderson which, by a simple modification of an algorithm in [CEI96, BP96] for constructing proofs, leads to a simple proof of the latter bound.

For the purposes of this abstract, we concentrate on the most interesting case for our results, namely that of random 3-CNF formulas.

2 Preliminaries

For the purpose of generating test formulas, the most natural model of a random k-CNF formula on n variables with clause density Δ is to choose $m = \Delta n$ clauses independently with replacement. This distribution, which we denote $\mathcal{F}_m^{k,n}$, is the one analyzed in [CS88]. Another model, which is used in [Fri], is to choose to include each of the possible clauses independently with probability $p = m/\binom{n}{k}2^k$; call this $\mathcal{F}^{k,n}(p)$. An easy argument shows that when considering properties that are monotone (or anti-monotone) with respect to sets of clauses, the almost certain properties under both distributions are the same up to a change from m to $m \pm o(m)$. This is just a natural extension of the similar (and more precise) equivalences for the random graph model as shown for example in [AV79]. We generally assume the distribution $\mathcal{F}_m^{k,n}$ but will sometimes switch to the $\mathcal{F}^{k,n}(p)$ distribution to simplify analysis. Also, when $k = 3$ we typically omit the superscript k.

In general, we write $F \sim \mathcal{F}$ to mean F is a random formula selected according to distribution $\mathcal{F}$.

We make frequent use of two well-known tail bounds for the binomial distribution (see [ASE92], Appendix A). If Y is a random variable distributed according to the binomial distribution $B(n,p)$ then

$$\Pr[Y < np/4] \leq 2^{-(np)/2} \quad (1)$$

$$\Pr[Y > Cnp] \leq (\frac{e}{C})^{-Cnp} \quad (2)$$

Following usual parlance we call partial assignments to variables *restrictions* and apply them in the usual way, writing $F\lceil_\rho$ for the result of applying ρ to F. We often wish to refer to the underlying set of literals set to true by a restriction ρ or contained in some clause C. We will abuse notation and write ρ or C for these sets of literals, respectively. We use $v(\rho)$, $v(C)$ to refer to the set of variables that underly ρ, C respectively and extend this in a natural way to sets of clauses.

A *resolution refutation* of a k-CNF formula F is a sequence of clauses, where each clause is either a clause from F, or follows from two previous clauses by the resolution (cut) rule — from $A \vee x$ and $B \vee \neg x$, derive $cut_x(A \vee x, B \vee \neg x) = A \vee B$ — and the final clause is the empty (false) clause, Λ. The size of a resolution refutation is the number of clauses in the proof. Each resolution refutation can be thought of as a natural directed acyclic graph whose nodes are labelled by the clauses, with each source node labelled by an input clause, with each non-source node of in-degree 2 labelled by the resolvent of its predecessors, and with a single sink node labelled by Λ.

The most widely used satisfiability algorithms are commonly called Davis-Putnam procedures, but, in fact, these procedures are derived from a system devised by Davis, Logemann and Loveland [DLL62] and so we will refer to them as DLL procedures. A DLL procedure is a form of recursive search for a satisfying assignment which on input F operates as follows: If F contains the empty clause Λ, it terminates reporting failure. Otherwise, a variable x is chosen and the procedure is applied recursively to $F\lceil_{x\leftarrow 0}$ and $F\lceil_{x\leftarrow 1}$. In choosing x, we always choose a variable that appears in some unit clause (clause of size 1) of F if one exists.

The recursive calls of the DLL procedure naturally induce a binary tree, each of whose internal nodes is labelled by a variable, with the out-edges of a node labelled by the 2 possible assignments to its associated variable. Each path in the tree corresponds to a partial assignment (restriction) and, if F is unsatisfiable, each leaf is labelled by a clause of F that becomes an empty clause under the restriction corresponding to the path to the leaf. This execution tree is a DLL refutation of F and its size is the number of recursive calls of the procedure on F.

It is not hard to show that the DLL refutation tree can be converted into a resolution proof of F of no larger size by working upwards from the clauses labelling the leaves and replacing each node labelled x by a cut on x.

If a node in a DLL tree is labelled by a variable x chosen because of a unit clause C then one child will falsify C so we can immediately proceed with the other recursive call. This is called *unit propagation*. When no unit clauses are present there may be different strategies for choosing the next variable to branch upon. Much of the work on DLL algorithms has been on refining the heuristics, called splitting rules, for making this choice. Usually there are trade-offs between the size of the proof produced and the time spent making this decision. One extremely simple splitting rule is the *ordered DLL procedure* [Mit95], used in the empirical studies of [SML96], where one begins with a fixed variable ordering and always chooses the next unset variable in that ordering.

3 Upper Bounds

In this section we show that two different algorithms, when run on 3-CNF formulas selected from F_m^n, produce resolution refutations of size $2^{O(n^2/m)}$ with high probability. Both algorithms work with respect to a fixed ordering, $x_1, x_2, \ldots, x_n$ of the variables. One is ordered DLL, the other, called Algorithm A, is more complicated but simpler to analyze. The analysis generalizes to k-CNF formulas.

Algorithm A. Let $t = 8n^2/m$. If $t > n/10$ then run ordered DLL. If $t \leq n/10$, run ordered DLL recursively until $x_1, \ldots, x_t$ are all assigned values. Let ρ denote the assignment along the current path, and $C_2(F,\rho)$ denote the set of clauses of size 2 in $F\lceil_\rho$. Run the (polynomial-time) algorithm for 2-SAT with input $C_2(F,\rho)$. The algorithm succeeds (finds a resolution refutation of F) if $C_2(F,\rho)$ is unsatisfiable for each ρ.

To analyze Algorithm A we need:

Lemma 1: Let F be a random 2-CNF formula chosen from $\mathcal{F}_{2n'}^{2,n'}$. Then the probability that F is satisfiable is $o(2^{-n'/9})$.

Proof Observe that the expected number of satisfying assignments for a 2-CNF formula with m' clauses and n' variables is $2^{n'}(3/4)^{m'}$ which is $o(2^{-n'/9})$ for $m' > 2.678n'$. (This bound can be reduced below 2 by using the techniques of [KKK96].) $\square$

Theorem 2: Let Δ be at least the threshold $c_k(n)$ and $m = \Delta n$. If $F \sim \mathcal{F}_m^n$, then, with probability $1-o(1)$ in n, Algorithm A produces a resolution refutation in time $2^{O(n^2/m)}n^{O(1)}$.

Proof Algorithm A clearly runs in time $2^{O(n^2/m)}n^{O(1)}$. Let $t = 8n^2/m$ and assume without loss of generality that $t < n/10$. To see that Algorithm A almost certainly yields a refutation of F, we show that almost certainly, for all assignments ρ to $\{x_1,\ldots,x_t\}$, $C_2(F,\rho)$ is unsatisfiable. (Other variables may be set by unit propagation but this only would improve the situation.)

Fix ρ and consider the size of $C_2(F,\rho)$ for random F. This size is a binomial random variable $B(m,q)$ where q is equal to the probability, for a random 3-clause C, that $C\lceil_\rho$ is a 2-clause:

$$\begin{aligned}
&\Pr[C\lceil_\rho \text{ is a 2-clause}]\\
&\quad= \quad 1/2\cdot\Pr[|v(\rho)\cap v(C)| = 1]\\
&\quad= \quad \frac{1}{2}\cdot\left(\frac{3t}{n} - \Pr[|v(\rho)\cap v(C)| > 1]\right)\\
&\quad= \quad \frac{1}{2}\cdot\left(\frac{3t}{n} - \frac{3(n-t)\binom{t}{2}+\binom{t}{3}}{\binom{n}{3}}\right) \geq t/n
\end{aligned}$$

Using (1), it follows that $\Pr[|C_2(F,\rho)| \leq 2n] \leq 2^{-4n}$. By Lemma 1 and the fact that the clauses in $C_2(F,\rho)$ are distributed uniformly at random on the remaining $n' = n-t$ variables, $\Pr[C_2(F,\rho)$ is satisfiable $\mid |C_2(F,\rho)| > 2n]$ is $o(2^{-n'/9})$. Since there are 2^t choices for ρ, the total failure probability is $2^t\cdot(o(2^{-(n-t)/9})+2^{-n})$, which is $o(1)$ since $(n-t)/9 \geq t$ for $t \leq n/10$. $\square$

Next we consider ordered DLL. At a point in the execution of DLL, say that a variable is *critical* if setting that variable either to 0 or 1 and then applying unit propagation creates the empty clause. Thus, if the splitting rule chooses that variable the current branch will terminate simply by unit propagation.

A point in the execution of DLL corresponds to some restriction ρ. We give a sufficient condition for a variable to be critical in terms of the set $C_2(F,\rho)$ of induced 2-clauses on the remaining set of n' variables. Define the standard directed graph $G(F,\rho)$ on $2n'$ vertices, one for each literal, that has directed edges $(\neg x, y)$, and $(\neg y, x)$ corresponding to each 2-clause $(x\vee y)$ in $C_2(F,\rho)$. It is easy to see that a sufficient condition for the variable x_i to be critical is that there be directed paths from x_i to $\neg x_i$ and from $\neg x_i$ to x_i, i.e., that x_i and $\neg x_i$ lie in the same strongly connected component.

Lemma 3: There exists a constant c such that if $F \sim \mathcal{F}_m^n$ and ρ is a fixed restriction of t variables with $n/2 \geq t \geq cn^2/m$, then with probability at least $1-2^{-n}$, for at least half of the $n' = n-t$ variables x_i of $C_2(F,\rho)$, x_i and $\neg x_i$ belong to the same strongly connected component of $G(F,\rho)$.

Proof Clearly, it suffices to show that with probability at least $1-2^{-n}$, $G(F,\rho)$ has a strongly connected component of size at least $3n'/2$. Let $C_1, C_2, \ldots, C_k$ be the strongly connected components ordered so that all edges between components go from lower to higher numbered components, and consider the first j such that $|C_1\cup\ldots\cup C_j| \geq n'/4$. We will show that the probability that $|C_j| < 3n'/2$ is at most 2^{-n}. If $|C_j| < 3n'/2$ then the set $S = C_1\cup\ldots\cup C_j$ satisfies $n'/4 \leq |S| \leq 7n'/4$ and there is no edge from $\bar{S}$ to S.

So to upper bound the probability that $|C_j| < 3n'/2$ it suffices to upper bound the probability that there is a set S with $n'/4 \leq |S| \leq 7n'/4$ which is *bad* in the sense that there is no edge from $\bar{S}$ to S. Fix S of size s, with $n'/4 \leq s \leq 7n'/4$. The probability that a randomly chosen 3 clause C, when restricted by ρ gives an edge from $\bar{S}$ to S is at least $s(n'-s-1)t/8\binom{n}{3} \geq \beta t/n$, for some constant $\beta > 0$. Hence the probability that none of the m clauses of F gives such an edge is at most $(1-\beta t/n)^m \leq e^{-\beta tm/n} \leq e^{-\beta cn} \leq 2^{-3n}$ for c chosen greater than 3β. There are at most $2^{2n'}$ such sets S, so the probability that there is a bad set S of size between $n'/4$ and $7n'/4$ is at most 2^{-n}. $\square$

Theorem 4: Let Δ be at least the threshold $c_k(n)$ and $m = \Delta n$. If $F \sim \mathcal{F}_m^n$, then ordered DLL will produce a resolution refutation for F in time $2^{O(n/\Delta)}n^{O(1)}$ with probability $1-o(1)$ in n.

Proof Without loss of generality we may assume that $m \geq 4cn$ where c is the constant of the previous lemma and let $t = cn^2/m$, so that $t \leq n/4$.

Fix a restriction ρ of the first t variables. We claim that the probability that there is a branch of the DLL tree consistent with ρ that is still active (not terminated) after the first $4t$ variables are set and the resulting unit propagations are processed is at most 2^{-2t}. Since there are 2^t choices for ρ, this will imply that with probability $1-2^{-t}$, every branch of ordered DLL is completed after at most the first $4t$ variables are fixed and all resulting unit propagations are done, and so the tree has at most $n2^{4t}$ nodes (including nodes from unit propagation).

To prove the claim, condition on the size r of the set of critical variables for $F\lceil_\rho$. By lemma 3, the probability that $r < n'/2$ is at most $2^{-n} \leq 2^{-4t}$, so we assume $r \geq n'/2$. The set of critical variables is equally likely to be any r-subset of the $n' = n-t$ unset variables, and so the probability that none of the next $3t$ variables in order are critical is at most $\binom{n'-3t}{r}/\binom{n'}{r} \leq (1-3t/n')^r \leq e^{-3t/2}$. Hence the probability that some branch consistent with ρ is unfinished after fixing the next $3t$ variables is at most $2^{-4t}+e^{-3t/2} \leq 2^{-2t}$. $\square$

4 Davis-Putnam lower bounds

In this section, we obtain lower bounds for certain natural classes of Davis-Putnam procedures. These lower bounds nearly match the upper bounds proven in the previous section and also provide a fairly precise analytical characterization of the behavior of the empirical results in [SML96].

Theorem 5: For $F \sim \mathcal{F}_m^n$, with probability $1-o(1)$, the size of the refutation produced by ordered DLL is $2^{\Omega(n^2/m)}$.

Proof Fix $t < \frac{n^2}{4m}$ and let S be the first t variables with respect to the given ordering. If ρ is an assignment to S such that $F\lceil_\rho$ has no Λ clause or unit clauses then the DLL tree for F will have a unique node corresponding to ρ. We will show that with probability $1-o(1)$ (as a function of t), $F \sim \mathcal{F}_n^m$ has the property that for at least $2^{t/2}$ of the 2^t assignments to ρ, that $F\lceil_\rho$ has no Λ clause or unit clauses. This implies that the DLL tree has size at least $2^{t/2}$, proving the theorem.

Let C_F^2 denote the set of maximal size subclauses of F of size two or three that are contained within S, and let $|C_F^2|$ denote the number of such clauses. The expected value of $|C_F^2|$ is $mt^2/n^2 < t/4$ by our bound on t. Call F *good* if (1) $|C_F^2| \leq t/2$ and (2) C_F^2 is satisfiable. The probability of (1) is $1-o(1)$ by tail bounds on the binomial distribution. Assuming (1) holds, C_F^2 consists of a collection of $\leq t/2$ *random* clauses of size two or three on set S of size t. Therefore the probability of (2) given (1) also goes to 1 because a random 2-CNF formula with $(1-\varepsilon)t$ variables on a set of size t is satisfiable almost certainly [Goe96, CR92]. (The fact that there are some 3-clauses as well as 2-clauses only makes C_F^2 easier to satisfy.) Thus, F is good with probability $1-o(1)$. Since there is a satisfying assignment for C_F^2, there must be a partial truth assignment setting at most $t/2$ variables that also satisfies C_F^2. For each of the at least $2^{t/2}$ assignments ρ obtained by extending this partial truth assignment to all the variables of S, $F\lceil_\rho$ will not contain any unit clauses or zero clauses (since all clauses in C_F^2 have been satisfied by ρ). $\square$

This lower bound is tight and we would like to extend it to a wider class of DLL procedures. The proof showed that along many paths in the DLL tree, unit clauses play no role. To generalize it, we first show that, for *any* DLL procedure, with probability $1-o(1)$ the number of variables set along any path by unit clause propagation is at most a small multiple of the number of variables set by splitting rules. More precisely, let F be a 3-CNF formula and let ρ be a partial truth assignment to some t variables. Then the unit clauses generated by ρ with respect to F are the unit clauses that we come across when setting ρ in F, and then setting all of the unit clauses that result from this restriction, and then setting all unit clauses resulting from this extended restriction, etc., until finally no more unit clauses are generated.

Lemma 6: Let $t > 0$ and $w = \max(t, \log_2 \binom{n}{t})$. There exists a constant c such that, for $F \sim \mathcal{F}_m^n$ with $m \leq cn^2/w$, the probability that there exists a partial assignment ρ to t variables such that ρ generates at least w unit clauses with respect to F is $o(1)$.

Proof We show that if F is a bad formula–that is, a formula such that there exists a partial assignment to t variables creating at least w unit clauses, then F can be described very succinctly. Thus, the total number of bad F's must be small.

We view an arbitrary formula $F \sim \mathcal{F}_m^n$ as an ordered sequence of m clauses. Thus F is uniformly chosen from $(8\binom{n}{3})^m$ possible formulas. Suppose there is a partial truth assignment ρ of size t that generates at least w unit clauses with respect to F. To encode F, we first encode ρ, the clauses of F that generate these w unit clauses, as well as their positions in F, and show that the number of such encodings is significantly smaller than $(8\binom{n}{3})^w$; the remaining $m-w$ clauses are encoded as before so we ignore them in calculating the savings.

Let $y_1,\ldots,y_t$ be the literals set to true by ρ in order of increasing variable indices. Define $y_{t+1},\ldots,y_{t+w}$ to be the literals appearing in the w unit clauses in the order they were generated (when several were generated by a single assignment, break ties arbitrarily). There are only $\binom{n}{t}n^w 2^{t+w}$ possibilities for these literals.

Call the clauses of F that become unit clauses under ρ, $C_1,\ldots,C_w$, where C_i is the clause corresponding to the literal y_{t+i}. The two remaining literals of each C_i are negations of literals in $\{y_1,\ldots,y_{t+i-1}\}$. Let the indices of the variables in the two literals be denoted by r_i and s_i, $r_i < s_i$. Since a unit clause is generated as soon as two literals in a clause of F have been falsified, it follows from the order in which we chose the literals that $1 \leq s_1 \leq \ldots \leq s_w < t+w$. Thus, the total number of choices for specifying all of the s_i's is at most $\binom{t+2w}{w}$. In a more straightforward manner, the total number of choices for the r_i's is at most $(t+w)^w$. Finally, there are at most m^w possible positions of these clauses in F.

Therefore the fraction of such F is at most

$$\binom{n}{t}n^w 2^{t+w}\binom{t+2w}{w}(t+w)^w m^w/(8\binom{n}{3})^w \;\leq\; \binom{n}{t}(cmw/n^2)^w$$

for some constant $c > 0$ since $w \geq t$. For $w \leq n^2/(3cm)$ this probability is $o(1)$ in n since $w \geq \log_2\binom{n}{t}$ and is $\omega(1)$ in n. $\square$

We say that a DLL procedure is *oblivious* if its splitting rule only depends on (1) the sequence of literals along the current path, and (2) the set of those clauses of F that only involve these variables. It is easy to argue that, for random formulas, the expected size of the refutation produced by any oblivious DLL algorithm is the same as for ordered DLL. However, this does not rule out the possibility that one such algorithm may be better than others on most inputs. Using Lemma 6, we can produce lower bounds for the more general class of oblivious DLL algorithms that are almost the same as those of Theorem 5 for large values of m.

Theorem 7: For any $\varepsilon > 0$ with probability $1-o(1)$ in n, any oblivious DLL procedure for a random m clause 3-CNF formula requires size $2^{\Omega(n^2/m\log n)}$ if $m > n^{3/2}\log^{\varepsilon/2} n$, and size $2^{\Omega(n/m^{1/3}\log^{1+\varepsilon} n)}$ for $m < n^{3/2}\log^{\varepsilon/2} n$.

Proof We can define the *branching depth* of a node v in a DLL refutation tree to be the number of degree 2 nodes on the path from the root of the tree to v after pruning any leaf that could be avoided by unit propagation.

Fix a formula F. Modify the oblivious DLL algorithm so that it the ignores the empty clause stopping rule. Truncate each branch when all the unit propagations have been done after a node of branching depth t is reached. Observe that we obtain a tree which contains precisely 2^t leaves of branching depth t. We want to show that most of these paths do not terminate early—i.e., most paths at branching depth t have not yet set F to false. By Lemma 6 we may assume without loss of generality that F is good in that all paths of branching depth t are of length at most $t+w$ where $w = \max(t, \log_2 \binom{n}{t})$. Fix such a path of length at most $t+w$.

Observe that all unit clauses encountered along the path have been satisfied and that all other clauses have not been used in the choice of branching decisions or examined in any other way. Therefore, other than the fact that they are not unit clauses at the time that the leaf is reached, we know nothing about them. This only rules out a small number of potential clauses and we see that the probability that the formula is set to false by this path is at most the expected number of such clauses set to false which is at most $m(t+w)^3/n^3$. Therefore, the expected number of paths that are falsified after t levels of branchings is at most $2^t m(t+w)^3/n^3$. For $m(t+w)^3/n^3 \le 1/\log^{\varepsilon} n$, the expected number of paths that are falsified after t levels of branching is at most $2^t/\log^{\varepsilon} n$. Thus by Markov's inequality, the probability that there are more than $2^t/2$ paths that are falsified after t levels of branchings is at most $2/\log^{\varepsilon} n$. Thus, with probability going to 1 as n goes to infinity, the proof size is at least $2^t - 2^t/2 = 2^t/2$.

Note that we required $m \le c'n^2/w$ for Lemma 6, and $m \le n^3/(t+w)^3 \log^{\varepsilon} n$ in the above calculation. Thus, we obtain a $2^t/2$ bound if $m < min(c'n^2/w, n^3/(t+w)^3 \log^{\varepsilon} n)$. When $w = \sqrt{n/c' \log^{\varepsilon/2} n}$, these two quantities are roughly equal. Using $t \le w \le t \log n$, we obtain a lower bound of $2^{\Omega(n^2/m \log n)}$ when $m \ge n^{3/2} \log^{\varepsilon/2} n$ and a lower bound of $2^{\Omega(n/m^{1/3} \log^{1+\varepsilon n})}$ when $m < n^{3/2} \log^{\varepsilon/2} n$. $\square$

For $m > n^{3/2} \log^{\varepsilon/2} n$, the lower bound from Theorem 7 is very nearly tight but below this value it weakens substantially. In the next section, we consider the much more general class consisting of all resolution proofs. We give lower bounds for these more general proofs that are actually larger than those of Theorem 7 bounds for values of m near n, in particular for $m < n^{12/11-\varepsilon}$.

5 General Resolution Lower Bounds

Let F be a Boolean formula. If F is unsatisfiable, define $\text{res}(F)$ to be the minimum number of clauses in a shortest resolution refutation of F. If F is satisfiable, define $\text{res}(F)$ to be ∞. The results of the previous section imply that for $F \sim \mathcal{F}_m^n$, $\text{res}(F) \le 2^{n(n/m)} n^{O(1)}$ with probability $1 - o(1)$ for m above the satisfiability threshold. In this section we prove lower bounds of a similar form:

Theorem 8: For each $\gamma > 0$, there exists a constant a_γ such that for all $m \ge n$, if $F \sim \mathcal{F}_m^n$ then with probability $1 - o(1)$, $\text{res}(F) \ge 2^{a_\gamma n (n/m)^{4+\gamma}}$.

Note that for $m = o(n^{5/(4+\gamma)})$, the bound is non-trivial, and for $m = O(n)$, the bound is $2^{\Omega(n)}$. Thus it includes the Chvátal-Szemerédi bound and it uniformly improves the Beame-Pitassi bounds. Our proof of this bound is fairly complicated, but we can obtain a somewhat weaker result (with the 4 replaced by a 5) via a significantly simpler argument. A key idea in our proof and that of [BP96] (which originates in [CS88]) is a result that provides a condition on a formula F so that any resolution proof of F has at least one large clause. We describe this in the next subsection.

5.1 Guaranteeing a Large Clause

A set of clauses is *b-bounded* if all clauses in the set have at most b literals. A formula F or a resolution proof $\mathcal{P}$ is *b-bounded* if its underlying set of clauses is b-bounded. For a real number σ, a set of clauses $\mathcal{C}$ is *σ-sparse* if $|\mathcal{C}| \le \sigma |v(\mathcal{C})|$ where $v(\mathcal{C})$ is the set of variables appearing in $\mathcal{C}$. For $s \ge 1$ and $\varepsilon \in (0,1)$, the following properties are defined for formulas F:

> Property $A(s)$: Every set of $r \le s$ clauses of F is 1-sparse.
>
> Property $B_\varepsilon(s)$: For r satisfying $s/2 < r \le s$, every set of r clauses of F is $\frac{2}{k+\varepsilon}$-sparse where k is the size of the largest clause of F.

The following result is essentially due to Chvátal and Szemerédi.

Proposition 9: Let $s > 0$ be an integer and F be a CNF formula. If properties $A(s)$ and $B_\varepsilon(s)$ both hold for F, then F has no $\varepsilon s/2$-bounded proof.

Proof The result holds trivially if F is satisfiable, so assume that F is unsatisfiable and let $\mathcal{P}$ be a resolution proof. View $\mathcal{P}$ as a DAG and for each clause (vertex) of $\mathcal{P}$, define $\text{sources}(D) = \text{sources}_{\mathcal{P}}(D)$ to be the set of those clauses of F that have a path to D. Define the boundary of a set $\mathcal{C}$ of clauses, $\beta(\mathcal{C})$ to be the set of variables that appear in exactly one clause of $\mathcal{C}$. It is easy to see that for any clause $D \in \mathcal{P}$, D contains $\beta(\text{sources}(D))$. We will prove two claims: (i) F satisfies $A(s)$ implies that there is a clause $C \in \mathcal{P}$ with $s/2 < |\text{sources}(C)| \le s$. (ii) F satisfies $B_\varepsilon(s)$ implies that for any subset $\mathcal{C}$ of clauses of F with $s/2 < |\mathcal{C}| \le s$,

$|\beta(\mathcal{C})| \geq \varepsilon|\mathcal{C}|$. Applying (ii) with $\mathcal{C} = sources(C)$ where C comes from (i), we conclude that $|C| > \varepsilon s/2$, as required.

To prove (i), we first note that if $\mathcal{C}$ is a set of clauses such that any subset is 1-sparse, then it is satisfiable. Indeed, the sparsity condition is equivalent to the hypothesis of the Hall theorem on systems of distinct representatives and the conclusion of the theorem is that there is a one-to-one mapping sending each clause $C \in \mathcal{C}$ to a variable $v_C \in C$. We can thus satisfy each clause C by appropriately fixing v_C.

sources(Λ) is an unsatisfiable set of clauses (since Λ is derived from it), so $A(s)$ implies $|\text{sources}(\Lambda)| > s$. Among clauses $D \in \mathcal{P}$ with sources$(D) > s$ choose a clause D' with $|\text{sources}(D')|$ minimum. Consider the two clauses C_1, C_2 that have edges into D'; clearly $|\text{sources}(D')| \leq |\text{sources}(C_1)| + |\text{sources}(C_2)|$ and hence one of C_1 or C_2 satisfies the requirement of C in (i).

To prove (ii), consider a subset of clauses $\mathcal{C}$, and let B, N be the sets of boundary and nonboundary variables, respectively. Let k be the size of the largest clause in F. The sum of the sizes of all the clauses is at most $k|\mathcal{C}|$ and is also at least $|B| + 2|N| = 2|v(\mathcal{C})| - |B|$. Hence $|B| \geq 2|v(\mathcal{C})| - k|\mathcal{C}|$. Since $s/2 < |\mathcal{C}| \leq s$, property $B_\varepsilon(s)$ implies $|v(\mathcal{C})| \geq (k+\varepsilon)|\mathcal{C}|/2$, and hence $|B| \geq \varepsilon s/2$. $\square$

5.2 The formula augmentation approach

We now give a simple proof of a weakening of Theorem 8:

Theorem 10: For each $\gamma > 0$, there exists a constant a_γ such that for all $m \geq n$, if $F \sim \mathcal{F}_m^n$ then with probability $1 - o(1)$, $\text{res}(F) \geq 2^{a_\gamma n(n/m)^{5+\gamma}}$.

The approach closely resembles the random restriction approach in [BP96], with one big difference: instead of applying a random restriction to F, we *augment* F by adding random clauses.

We say that a clause C *subsumes* a clause D if $C \subset D$. Suppose that $\mathcal{P}$ is a proof of F and let $F \wedge G$ be an augmentation of F where G is a CNF formula. We can obtain a proof of $F \wedge G$, which we call $\mathcal{P}\lceil_G$, as follows: For each clause $D \in \mathcal{P}$, if there is a clause C in G such that C subsumes D, remove the part of the derivation that produced D and replace D by C. Propagate this simplification through the rest of the proof by (possibly) shortening clauses that were produced using D.

Suppose now that F is a "suitably sparse" function that has a "small" proof $\mathcal{P}$. If we apply a random augmentation G from some suitable distribution $\mathcal{G}$, then, because the proof is small, with high probability G will subsume all "large enough" clauses in $\mathcal{P}$ and hence $\mathcal{P}\lceil_G$ will be a proof for $F \wedge G$ with no large clauses. If $F \wedge G$ is suitably sparse with high probability, then we obtain a contradiction by applying Proposition 9 to say that $F \wedge G$ has no such proof. This motivates what follows.

We say that $\mathcal{G}$ satisfies property $g(b,M)$, for $b,M > 0$ if for any clause C of size at least b, the probability that $G \sim \mathcal{G}$ does not subsume C is at most $1/M$. We now formulate a criterion for res(F) to be large:

Proposition 11: Let F be a k-bounded formula. Let $s,M \geq 1$ and $\varepsilon > 0$, and let $\mathcal{G}$ be a distribution over CNF formulas that satisfies $g(\varepsilon s/2, M)$. Then $\text{res}(F) \geq$
$M \times \Pr_{G\sim\mathcal{G}}[F \wedge G$ satisfies both $A(s)$ and $B_\varepsilon(s)]$.

Proof The conclusion follows immediately from the chain of inequalities:

$$\begin{aligned}
&\text{res}(F)/M \\
&\quad \geq \Pr_{G\sim\mathcal{G}}[\mathcal{P}\lceil_G \text{ is not } \varepsilon s/2 \text{ bounded}] \\
&\quad \geq \Pr_{G\sim\mathcal{G}}[F \wedge G \text{ satisfies both } A(s) \text{ and } B_\varepsilon(s)].
\end{aligned}$$

The second inequality is immediate from Proposition 9. For the first inequality, if $\mathcal{P}$ is a proof of F of size res(F), it has at most res(F) clauses of size at least $\varepsilon s/2$, and by property $g(\varepsilon s/2, M)$ the probability that G does not subsume all such clauses is at most res$(F)/M$. $\square$

For distributions over formulas we have:

Theorem 12: Let $\mathcal{F}$ be a distribution over k-bounded formulas. Let $s,M \geq 1$ and $\varepsilon > 0$ and suppose that $\mathcal{G}$ is a distribution over formulas that satisfies $g(\varepsilon s/2, M)$. Then

$$\begin{aligned}
&\Pr_{F\sim\mathcal{F}}[res(F) < M/2] \\
&\quad \leq 2(\Pr_{F\sim\mathcal{F},G\sim\mathcal{G}}[F \wedge G \text{ does not satisfy } A(s)\ ] \\
&\qquad + \Pr_{F\sim\mathcal{F},G\sim\mathcal{G}}[F \wedge G \text{ does not satisfy } B_\varepsilon(s)]).
\end{aligned}$$

Proof For a formula F, let $p(F)$ denote the sum of $Pr_G[F \wedge G$ does not satisfy $A(s)]$ and $Pr_G[F \wedge G$ does not satisfy $B_\varepsilon(s)]$. By Proposition 11, $\text{res}(F) > (1 - p(F))M$. Thus $\Pr_F[\text{res}(F) < M/2] \leq \Pr_F[p(F) > 1/2] < 2E_F[p(F)]$, where $E[\cdot]$ denotes expectation. This last quantity is equal to the right hand side of the claimed inequality. $\square$

We apply this Theorem to the case $\mathcal{F} = \mathcal{F}_m^n$ by choosing $\mathcal{G} = \mathcal{F}$. Note that $F \wedge G \sim \mathcal{F}_{2m}^n$. We first want to argue that $F \wedge G$ satisfies both $A(s)$ and $B_\varepsilon(s)$ almost certainly for some suitable s.

Lemma 13: For any ε with $0 < \varepsilon < 1$ there is a constant $c_\varepsilon > 0$ such that the following holds. Let $F \sim \mathcal{F}_m^n$, where $m \geq n$. If $s \leq c_\varepsilon n(n/m)^{2/(1-\varepsilon)}$ then F satisfies both $A(s)$ and $B_\varepsilon(s)$ with probability $1 - o(1)$ in s.

This lemma is a special case of a lemma proved in [BP96]. (Note also that Lemma 16 below strengthens this.) Applying this lemma with m replaced by $2m$ we conclude that for any $\varepsilon > 0$ and for some constant $b_\varepsilon > 0$, if $s \leq b_\varepsilon n(n/m)^{2/(1-\varepsilon)}$ then $F \wedge G$ satisfies $A(s)$ and $B_\varepsilon(s)$ with probability $1 - o(1)$. Next, we want to determine as large an M as possible so that $\mathcal{G}$ satisfies $g(\varepsilon s/2, M)$. For a clause C of size at least $\varepsilon s/2$, there are $8\binom{\varepsilon s/2}{3}$ clauses of size 3 that subsume C. If $G \sim \mathcal{G}$, then the probability that none of its m clauses subsume C is at most $(1-\binom{\varepsilon s/2}{3}/\binom{n}{3})^m \leq 2^{-b_\varepsilon ms^3/n^3}$ for some constant b_ε. Substituting $s = b_\varepsilon n(n/m)^{2/(1-\varepsilon)}$, we have that, for sufficiently small ε, $\mathcal{G}$ satisfies $g(\varepsilon s/2, 2^{d_\varepsilon n(n/m)^{5+7\varepsilon}})$ for some constant d_ε. Hence with probability $1-o(1)$, $\mathrm{res}(F) \geq 2^{\Omega(n(n/m)^{5+\gamma})}$ for any $\gamma > 0$.

An Open Problem The upper bound on s in the statement of Lemma 13, as we show in proving the more general Lemma 16 below, is needed for the analysis of property $B_\varepsilon(s)$. For property $A(s)$, $s \leq cn^2/m$ suffices. As far as we know, property $B_\varepsilon(s)$ may also hold for some $s = \Omega(n^2/m)$. Showing this would improve the bound in Theorem 8 to $2^{\Omega(n^3/m^2)}$.

5.3 The Deletion Argument

In this subsection, we present another variant of the Beame-Pitassi approach, which leads to the bound stated in Theorem 8. Their argument used restrictions instead of augmentation to remove large clauses. Given a resolution proof $\mathcal{P}$ and a restriction ρ, a resolution proof $\mathcal{P}\lceil_\rho$ is obtained by replacing each clause $C \in \mathcal{P}$ by $C\lceil_\rho$ and then, removing any clause $C\lceil_\rho$ that is set to 1, and contracting to edges all paths in the proof whose internal nodes have in-degree 1. It is easy to verify that if $\mathcal{P}$ is a resolution proof for F then $\mathcal{P}\lceil_\rho$ is a resolution proof for $F\lceil_\rho$. Observe that a restriction ρ corresponds naturally to an augmentation G with only unit clauses, and there is a close correspondence between the proofs $\mathcal{P}\lceil_G$ and $\mathcal{P}\lceil_\rho$.

If $\mathcal{R}$ is a probability distribution over restrictions, we say that $\mathcal{R}$ has *property* $R(b,M)$ for $b,M > 0$ if for any clause C on X of size at least b, $\Pr[\rho \text{ does not satisfy } C] \leq 1/M$.

Theorem 14: Let $\mathcal{F}$ be a distribution over k-bounded formulas. Let s, $M \geq 1$ and $\varepsilon > 0$ and suppose that $\mathcal{R}$ is a distribution over restrictions that satisfies $R(\varepsilon s/2, M)$. Then

$$\begin{aligned}\Pr_{F\sim\mathcal{F}}&[\mathrm{res}(F) < M/2]\\ &\leq 2(\Pr_{F\sim\mathcal{F},\rho\sim\mathcal{R}}[F\lceil_\rho \text{ does not satisfy } A(s)\,]\\ &\quad + \Pr_{F\sim\mathcal{F},\rho\sim\mathcal{R}}[F\lceil_\rho \text{ does not satisfy } B_\varepsilon(s)\,]).\end{aligned}$$

While not stated explicitly in [BP96] this theorem captures the essence of their approach. We now sketch its application to lower bounding $\mathrm{res}(F)$ for $F \sim \mathcal{F}^n_m$. Let $\mathcal{R}_t$ denote the distribution over restrictions where we first choose $v(\rho) \subseteq X$ by selecting each variable independently with probability t/n and then set the selected variables uniformly at random. Applying Theorem 14 involves two steps. The first step is a generalization of Lemma 13 which shows that $A(s)$ and $B_\varepsilon(s)$ also hold almost certainly for $F\lceil_\rho$ for $F \sim \mathcal{F}^n_m$, $\rho \sim \mathcal{R}_t$, and $t \leq \sqrt{n}(n/m)^{1/2+\varepsilon}$. The second step is to determine as large an M as possible such that the distribution $\mathcal{R}_t$ satisfies $R(\varepsilon s/2, M)$, where s and t are the largest numbers for which the previous lemma applies. Once this is done, Theorem 14 immediately implies that with probability $1-o(1)$, $\mathrm{res}(F) \geq M/2$. Now, for a fixed clause C and for $\rho \sim \mathcal{R}_t$, a variable x that is in C is fixed by ρ to satisfy C with probability $t/2n$ so the probability that ρ doesn't satisfy a given clause C is at most $(1-t/2n)^{|C|} \leq e^{-|C|t/2n}$. Thus, $\mathcal{R}_t$ satisfies $R(\varepsilon s/2, e^{\varepsilon st/4n})$. Applying Theorem 14 gives that for any $\gamma > 0$, $\mathrm{res}(F) \geq 2^{\Omega(n^{1/2}(n/m)^{5/2+\gamma})}$ with probability $1-o(1)$ in n.

A major bottleneck in the Beame-Pitassi argument is the upper bound on the size t of the restriction needed to extend Lemma 13. Indeed, for t much larger than $n/\sqrt{m}$, there is a substantial probability that $A(s)$ fails for $F\lceil_\rho$. For example, an easy computation shows that the probability $F\lceil_\rho$ has no empty clauses is $e^{-\Theta(mt^3/n^3)}$ and since the presence of an empty clause in $F\lceil_\rho$ violates $A(s)$, we have that $t = o(n/m^{1/3})$. (Considering the effect of the unit clauses created by the restriction constrains t even further).

To overcome this limitation, we want to avoid the creation of clauses of size 0 or 1 in $F\lceil_\rho$. We do this by allowing our restriction distribution to depend on F. Thus for each formula F, we will specify a distribution $\mathcal{R}(F)$ over restrictions, tailored to F so as to avoid the creation of clauses of size 0 or 1. This will complicate the argument because we lose independence both between $\mathcal{F}$ and ρ and also within ρ. Before defining the restriction distributions, we give an easy generalization of Theorem 14 to handle this situation.

Theorem 15: Let $\mathcal{F}$ be a distribution over formulas. Suppose that for each formula F, $\mathcal{R}(F)$ is a distribution over restrictions and let s, $M \geq 1$ and $\varepsilon > 0$. Then:

$$\begin{aligned}&\Pr_{F\sim\mathcal{F}}[res(F) < M/2]\\ &\quad\leq\; 2\times(\Pr_{F\sim\mathcal{F},\rho\sim\mathcal{R}}[F\lceil_\rho \text{ does not satisfy } A(s)\,]\\ &\qquad + \Pr_{F\sim\mathcal{F},\rho\sim\mathcal{R}}[F\lceil_\rho \text{ does not satisfy } B_\varepsilon(s)\,])\\ &\qquad + \Pr_{F\sim\mathcal{F}}[\mathcal{R}(F) \text{ does not satisfy } R(\varepsilon s/2, M)]\end{aligned}$$

Instead of proving our lower bound with respect to the distribution $\mathcal{F}^n_m$, we will prove it for the distribution $\mathcal{F}^n(p)$ with $p = m/8\binom{n}{3}$. As noted earlier, lower bounds on $\mathrm{res}(F)$ for the latter distribution imply essentially the same lower bounds for the former.

We now define the restriction distributions. First, we associate to a formula F a graph $\Gamma(F)$ on vertex set X with two variables joined by an edge if there is a clause of F that contains them both. We refer to the set of variables adjacent to $x \in X$ as the *F-neighbors* of x and the number of such variables as the F-degree of x. Given a 3-CNF formula F on n variables, and $t, U \in [0,n]$ we define the distribution $\mathcal{R}^U_t(F)$ over random restrictions as follows: First, choose a restriction $\rho^{binom} \sim \mathcal{R}_t$. Then, delete any assignment ρ^{binom} makes to variables with more than U F-neighbors, obtaining a restriction ρ^{small}. Finally, delete any assignment ρ^{small} makes to a variable that has any F-neighbor assigned a value by ρ^{small}. Then $\rho \sim \mathcal{R}^U_t(F)$ is the resulting restriction.

To apply Theorem 15 we upper bound, as a function of n, m, s, ε, t, U, M, the three probabilities that occur on the right hand side, and choose s, t, U, M so that all three probabilities are $o(1)$. The crux of the argument is contained in three main lemmas. The first of these lemmas upper bounds the probability that $F\lceil_\rho$ fails to satisfy $A(s)$ and $B_\varepsilon(S)$.

Lemma 16: Let $F \sim \mathcal{F}^n(p)$ and $\rho \sim \mathcal{R}_t^U(F)$.

1. There is a constant C_0 such that if $s,t \leq C_0 n(n/m)$ then, with probability $1-o(1)$ in s, $F\lceil_\rho$ satisfies $A(s)$.
2. For any ε with $0 < \varepsilon < 1$ there is a constant $c_\varepsilon > 0$ such that if $s,t \leq c_\varepsilon n(n/m)^{2/(1-\varepsilon)}$, with probability $1-o(1)$ as a function of s, $F\lceil_\rho$ satisfies property $B_\varepsilon(s)$.

Next we upper bound the probability with respect to F that $\mathcal{R}_t^U(F)$ fails to satisfy $R(\varepsilon s/2, M)$, by showing that for most F, and for any clause C of size $\varepsilon s/2$, if $\rho \sim \mathcal{R}_t^U(F)$, then ρ satisfies C with high probability. This reduces easily to showing that $|v(\rho) \cap v(C)|$ is large as follows. Say that a distribution $\mathcal{R}$ is *fair* if when conditioned on the set of variables assigned, it is uniform over the set of assignments to those variables. Note that the distributions $\mathcal{R}_t^U(F)$ are fair. For any clause C, conditioned on $|v(C) \cap v(\rho)| \geq w$, a fairly distributed ρ fails to satisfy C with probability at most 2^{-w}.

Say that $\mathcal{R}_t^U(F)$ satisfies $R^*(b,w)$ if for any $B \subseteq X$ of size at least b, and for $\rho \sim \mathcal{R}_t^U(F)$, the probability that ρ fixes fewer than w variables of B is at most 2^{-w}. A fair distribution that satisfies $R^*(b,w)$ is easily seen to satisfy $R(b, 2^{w-1})$.

So we will show that $\mathcal{R}_t^U(F)$ almost certainly satisfies $R^*(b,w)$ for some appropriate b,w.

Using a simple analysis as in [BP96], one can show that $\mathcal{R}_t$ has property $R^*(b,w)$, where w is Cbt/n for some constant C. We would like a similar result for $\mathcal{R}_t^U(F)$, and to do this we must show that not too many of the variables of B fixed by ρ^{binom} are unset in going to ρ. One crude way to do this is to show that with high probability at most $w/2$ variables overall are unset, and this can be done by suitably constraining t. Completing this analysis leads to a bound matching the one obtained in section 5.2.

To obtain the bound of Theorem 8 a subtler argument is necessary – one that takes advantage of the fact that for each B we only care about the way the deleted variable assignments occur in B. We first give a rough sketch. Based on F and a fixed set B, we classify the variables of F-degree at most U as bad or good, where a bad variable is one which, if assigned by ρ^{binom}, could cause many assigned variables inside B to be unset. The sets of bad and good variables are denoted $Bad_F(B)$ and $Good_F(B)$. We prove that a random F satisfies certain structural properties: not many variables have F-degree greater than U and for every B, not many variables are in $Bad_F(B)$. For fixed F satisfying these properties and any B, we show that, with exponentially small failure probability, the number of variables of $B \cap Good_F(B)$ having no bad neighbors that are set by ρ^{binom} is at least $|B|/4$. Conditioning on this, we can show that, with exponentially small failure probability, a constant fraction of the variables of $B \cap Good_F(B)$ that are set by ρ^{binom} remain set in ρ. This is enough to conclude that $R^*(b,w)$ is satisfied.

We now define the structural conditions on the formula F, and show that, for suitable values of the relevant parameters, any formula F that satisfies these conditions satisfies $R^*(b,w)$. We need some definitions. We say that a variable is *large* if its F-degree is more than U and is *small* otherwise. For a subset B of variables we define a set of variables $Bad_F(B)$ to consist of all small variables x such that either
(i) $x \notin B$ and x has at least 2 F-neighbors in B, or
(ii) $x \in B$ and x has at least 48 F-neighbors in B.
Any small variable x not in $Bad_F(B)$ is in $Good_F(B)$. We introduce three parameterized properties that may hold for a formula F:

F1(b,U): There are at most $b/4$ large variables.

F2(b): For all sets B of size b, B contains at most $b/4$ variables in $Bad_F(B)$.

F3(b,C): For all sets B of size b, $X \setminus B$ contains at most Cb variables in $Bad_F(B)$.

Lemma 17: For any constant $C > 0$ we can choose constants $\beta,\mu > 0$ so that the following holds. If n, b, t, U are positive integers satisfying $b,t \leq \beta n/U$ and F is a formula satisfying F1(b,U), F2(b) and F3(b,C), then $\mathcal{R}_t^U(F)$ satisfies $R^*(b,\mu bt)$ and hence also $R(b, 2^{\mu bt-1})$.

Lemma 18: Let $m \geq n$ and $F \sim \mathcal{F}^n(p)$ where $p = m/8\binom{n}{3}$.

1. For some constant $C_1 > 0$, if $b < n/2$ and $U > C_1(m/n)\log(n/b)$ then F satisfies F1(b,U) with probability at least $1-o(1)$ in b.
2. If $b < n(n/m)^2/20$, then F satisfies F2(b) with probability $1-o(1)$ in b.
3. If $\varepsilon_3 > 0$, $C_3(\varepsilon_3)$ is sufficiently large and $\alpha_3(\varepsilon_3) > 0$ is sufficiently small, then for $b < \alpha_3 n(n/m)^{2+\varepsilon_3}$, F satisfies F3(b, C_3) with probability $1-o(1)$ in b.

We can put this all together to prove Theorem 8. Given $\gamma \in (0,1)$, set $\varepsilon = \gamma/2$. Let $n,m \geq 0$ be given, and set $t = s = h_\varepsilon n(n/m)^{2+\gamma/2}$, where h_ε is suitably small (depending only on ε). Set $b = \varepsilon s/2$ and $U = C_1 n/m \log(n/b)$ where C_1 is suitably large. We apply Theorem 15 for $\mathcal{F}^n(p)$ and restriction distributions $\mathcal{R}_t^U(F)$. By Lemma 16, the probability that $F\lceil_\rho$ fails to satisfy $A(s)$ and $B_\varepsilon(s)$ is $o(1)$ as a function of s. For the given parameters, F satisfies F1(b,U), F2(b), F3(b,C_3) for some suitably large C_3 with probability $1-o(1)$ by Lemma 18. By Lemma 17 $\mathcal{R}_t^U(F)$ satisfies $R^*(b,\mu st/n)$ and hence $R(b, 2^{\mu st/n}-1)$ where μ is a suitably small constant. Thus, for suitably small a_γ and for $M = 2^{1+a_\gamma n(n/m)^{4+\gamma}}$, all three terms on the right hand side of the inequality in Theorem 15 are $o(1)$. Theorem 8 follows.

So it remains to prove the lemmas. Due to space considerations, we omit the proof of Lemma 18, noting that the proof is a generally straightforward exercise in combinatorial probability.

Proof of Lemma 16 We may view our probability space as the set of pairs (F,ρ^{binom}) where F is chosen according to $\mathcal{F}^n(p)$ and ρ^{binom} is chosen independent of F from $\mathcal{R}_t$; note that ρ is determined by (F,ρ^{binom}). We will condition on $v(\rho^{binom}) = Y$ for fixed $Y \subseteq X$, which does not bias the distribution of F. By standard Chernoff-Hoeffding bounds, $\Pr[|v(\rho^{binom})| > 2t] = e^{-\Theta(t)}$ which is $o(1)$ in n. Thus it suffices to show that for any fixed Y with $|Y| \le 2t$, the probabilities over $\mathcal{F}^n(p)$ that $F\lceil_\rho$ fails to satisfy $A(s)$ and $B_\varepsilon(s)$ with these parameters, conditioned on $v(\rho^{binom}) = Y$, is $o(1)$.

For integers r,q, let $Q(r,q)$ denote the probability (conditioned on $v(\rho^{binom}) = Y$) that there exists a set S of r variables that contains at least q clauses of $F\lceil_\rho$. Note that the probability that $A(s)$ does not hold for $F\lceil_\rho$ is at most $\sum_{r=2}^{s} Q(r,r+1)$ and the probability that $B_\varepsilon(s)$ does not hold is at most $\sum_{s/2\le r\le s} Q(r,\sigma r)$ where $\sigma = 2/(3+\varepsilon)$. We will upper bound these probabilities by upper bounding $Q(r,q)$.

Let $S \subseteq X$ with $|S| = r$. In order for a clause C of F to be contained in S after ρ is applied, it is necessary that either all of its variables are in S or two of its variables are in S and one is in Y and ρ sets that variable to the value that does not satisfy C. Let N denote the number of such clauses; trivially $N \le 8\binom{r}{3} + 4|Y|\binom{r}{2} \le 8\binom{r}{3} + 8t\binom{r}{2}$. The distribution of the number of clauses that are contained in S is then given by the binomial distribution, $B(N,p)$. The probability that at least q clauses of F are contained in S after ρ is applied is bounded above by:

$$\Pr[B(N,p) \ge q] \le \binom{N}{q} p^q \le \left(\frac{C_1 m r^2(t+r)}{qn^3}\right)^q, \qquad (3)$$

for some constant $C_1 > 0$. Since the number of r element subsets of X is $\binom{n}{r} \le (en/r)^r$, we conclude

$$Q(r,q) \le \left(\frac{ne}{r}\right)^r \left(\frac{C_1 m r^2(t+r)}{qn^3}\right)^q,$$

As noted above, the probability that $A(s)$ does not hold is at most $\sum_{r=2}^{s} Q(r,r+1)$. For $r \le s$ we have:

$$\begin{aligned} Q(r,r+1) &\le \left(\frac{ne}{r}\right)^r \left(\frac{C_1 m r^2(t+r)}{(r+1)n^3}\right)^{r+1} \\ &\le \left(\frac{C_2 m(s+t)}{n^2}\right)^r \frac{C_3 m r(s+t)}{n^3} \le \frac{r}{2^r n} \end{aligned}$$

for constants $C_2, C_3 > 0$, provided that $s,t \le C_0 n\frac{n}{m}$, for some appropriately chosen constant $C_0 > 0$. Thus the probability that $F\lceil_\rho$ does not satisfy $A(s)$ is at most $\frac{1}{n}\sum_{r=2}^{s} r/2^r < 3/2n$ which is clearly $o(1)$. This proves part 1.

To upper bound the probability that $F\lceil_\rho$ does not satisfy $B_\varepsilon(s)$, we first upper bound $Q(r,\sigma r)$ for $s/2 \le r \le s$:

$$\begin{aligned} Q(r,\sigma r) &\le \left(\frac{ne}{r}\right)^r \left(\frac{C_1 m r(t+r)}{\sigma n^3}\right)^{\sigma r} \\ &= \left(\frac{(ne)^{1/\sigma} C_1 m r(t+r)}{\sigma r^{1/\sigma} n^3}\right)^{\sigma r} \\ &\le \left(\frac{C_4 m r^{1-1/\sigma}(r+t)}{n^{3-1/\sigma}}\right)^{\sigma r} < 2^{-\sigma r} \end{aligned}$$

for appropriate constant C_4, provided that

$$\frac{C_4 m r^{1-1/\sigma}(r+t)}{n^{3-1/\sigma}} \le \frac{1}{2}.$$

For $\sigma = 2/(3+\varepsilon)$ this is satisfied if both s and t are at most $c_\varepsilon n \cdot (\frac{n}{m})^{2/(1-\varepsilon)}$, for some constant c_ε. Now the total failure probability for property $B_\varepsilon(s)$ is at most $\sum_{r=s/2}^{s} 2^{-\sigma r}$ which is clearly $o(1)$ for non-constant s and Part 2 is proved. □

Proof of Lemma 17 Fix F satisfying F1(b,U),F2(b) and $F_3(b,C)$ and fix $B \subseteq X$ of size b. We need to show that for some $\mu > 0$, $\Pr[|B \cap v(\rho)| \le \mu mbt/n] \le 2^{-\mu bt/n}$.

Let $S \subset X$ be the set of small variables, i.e. those having at most U F-neighbors. For the variables $x \in S$, $x \in v(\rho^{small})$ independently with probability t/n and for $x \notin S$, $x \notin v(\rho^{small})$. Also, given that $x \in v(\rho^{small})$, x will be in ρ if and only if none its at most U F-neighbors is in ρ^{small}. Thus if $x \in S$, $\Pr[x \in v(\rho)] \ge \frac{t}{n}(1-\frac{Ut}{n}) \ge \frac{t}{2n}$, where the last inequality requires $\beta \le 1/2$.

By F1(b,U), $|B \cap S| \ge 3b/4$ so $E[|v(\rho) \cap B|] \ge \frac{3bt}{8n}$. If the events $x \in v(\rho)$ were independent for $x \in B \cap S$, or if $B \cap S$ had a large (constant fraction) subset that was independent, we could use binomial tail bounds to reach the desired conclusion. Now, for a subset I of S, it is easy to see that for $x \in I$, the events $x \in v(\rho)$ will be independent if and only if no two variables of I are F-neighbors and no two variables in I share the same F-neighbor. In other words, I is an independent set of vertices with respect to the graph $\Gamma^2(F)$, where two vertices are adjacent if they are within distance 2 in $\Gamma(F)$. In general, unfortunately, there will not exist a large enough subset of independent variables in $B \cap S$. To overcome this problem, we will only consider variables in $B \cap Good_F(B)$ (which is contained in $B \cap S$) and condition on the choice of $\rho^{small}_{Bad_F(B)}$, i.e., the projection of ρ^{small} on the variables of $Bad_F(B)$.

Let $J \subseteq B \cap Good_F(B)$ be the set of variables in $B \cap Good_F(B)$ that have no F-neighbor in $v(\rho^{small}_{Bad_F(B)})$. By definition, variables in J do not share any F-neighbors in $Good_F(B) \setminus B$. Thus, conditioned on the choice of $\rho^{small}_{Bad_F(B)}$, the only dependence for $x,y \in J$, between the events $x \in v(\rho)$ and $y \in v(\rho)$ occurs if x and y are F-neighbors or if they share a common F-neighbor in $B \cap Good_F(B)$. Let $\Gamma'(F,B)$ denote

the graph induced by $\Gamma(F)$ on $B \cap Good_F(B)$. It follows that if $I \subset J$ is an independent set in the graph $(\Gamma')^2(F,B)$, then the events $x \in v(\rho)$ for $x \in I$ are indeed independent random variables (with respect to this conditional distribution). We now make two claims:
(i) The subgraph of $(\Gamma')^2$ induced on the set J has maximum degree less than 2500, and
(ii) $\Pr[|J| < b/4] \leq 2^{-bt/n}$.

Assuming these claims, we finish the proof. The probability that $|B \cap v(\rho)| \leq \mu bt/n$ is at most

$$\Pr_{\rho^{small}_{Bad_F(S)}}[|J| < b/4] + \Pr_{\rho^{small}_{Good_F(B)}}[|B \cap v(\rho)| \leq \mu bt/n \mid |J| \geq b/4]$$

The first probability is at most $2^{-bt/n}$ by the second claim. To bound the second probability, fix J with $|J| \geq b/4$. The choice of $\rho^{small}_{Good_F(B)}$ is independent of J since it is independent of $\rho^{small}_{Bad_F(B)}$. It is well known that a graph with v vertices and maximum degree Δ has an independent set I of size at least $v/(\Delta+1)$ (e.g., construct I by a sequential greedy algorithm). So there is an independent I of $(\Gamma')^2$ of size at least $b/10000$. Conditioned on $|J| \geq b/4$, the events $x \in v(\rho)$ for $x \in I$ is a set of mutually independent random variables with $\Pr[x \in v(\rho)] \geq \frac{t}{2n}$, so the expectation of their sum is at least $bt/20000n$. Using binomial tail bounds, if $\mu < 1/80000$, the conditional probability that $|I \cap v(\rho)| \leq \mu bt/n$ is at most $2^{-\mu bt/n-1}$. Thus $\Pr[|B \cap v(\rho)| \leq \gamma bt/n]$ is at most $2^{-\gamma bt/n-1} + 2^{-bt/n} \leq 2^{-\mu bt/n}$ as required.

It remains to prove the two claims.

For the first claim, fix a variable $x \in B \cap Good_F(B)$; we bound the number of $(\Gamma')^2$-neighbors of x in $B \cap Good_F(B)$. By definition, there are at most 48 F-neighbors of x in $B \cap Good_F(B)$, and each has at most 47 F-neighbors in $B \cap Good_F(B)$ other than x, giving an upper bound of $(48)^2 < 2500$ on the number of $(\Gamma')^2$-neighbors of x inside $B \cap Good_F(B)$.

For the second claim, note that $|J| \geq |B \cap Good_F(B)| - N$, where N is the total number of F-neighbors of $v(\rho^{small}_{Bad_F(B)})$. Now F1$(b,U)$ and F2(b) imply that $|B \cap Good_F(B)| \geq b - 2 \cdot b/4 = b/2$. Since each variable in $Bad_F(B)$ has at most U F-neighbors, $|N| \leq U \cdot |v(\rho^{small}_{Bad_F(B)})|$. Thus $|J| < b/4$ implies that $|v(\rho^{small}_{Bad_F(B)})| > b/(4U)$, so it suffices to upper bound the probability of the latter event. Now $|v(\rho^{small}_{Bad_F(B)})|$ is a sum of independent Bernoulli random variables with probability t/n. By F2(b) and F3(b,C), $|Bad_F(B)| \leq (C+1/4)b$, and so by binomial tail estimates we have

$$\Pr[|v(\rho^{small}_{Bad_F(B)})| > b/(4U)] \leq (8e(C+1/4)Ut/n)^{b/(4U)}.$$

Using $t \leq \beta n/U$ and assuming β is small enough (depending only on C), this is at most $(8e(C+1/4)\beta)^{bt/4\beta n} \leq 2^{-bt/n}$. This proves the claim and the lemma. $\square$

References

[ASE92] N. Alon, J. Spencer, and P. Erdos. *The Probabilisitic Method.* John Wiley and Sons, Inc., 1992.

[AV79] D. Angluin and L. Valiant. Fast probabilistic algorithms for Hamiltonian circuits and matchings. *Journal of Computer and System Sciences*, 18:155–193, 1979.

[BFU93] A. Broder, A. Frieze, and E. Upfal. On the satisfiability and maximum satisfiability of random 3-CNF formulas. January 1993.

[BP96] P. Beame and T. Pitassi. Simplified and improved resolution lower bounds. In *37th Annual Symposium on Foundations of Computer Science*, pages 274–282, Burlington, VT, October 1996. IEEE.

[CA96] J.M. Crawford and L.D. Auton. Experimental results on the crossover point in random 3SAT. *Artificial Intelligence*, 81, 1996.

[CEI96] M. Clegg, J. Edmonds, and R. Impagliazzo. Using the Gröbner basis algorithm to find proofs of unsatisfiability. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, pages 174–183, Philadelphia, PA, May 1996.

[CF90] M.T. Chao and J. Franco. Probabilistic analysis of a generalization of the unit-clause literal selection heuristics. *Information Science*, 51:289–314, 1990.

[CR92] V. Chvátal and B. Reed. Mick gets some (the odds are on his side). In *33nd Annual Symposium on Foundations of Computer Science*, pages 620–627, Pittsburgh, PA, October 1992. IEEE.

[CS88] V. Chvátal and Endre Szemerédi. Many hard examples for resolution. *Journal of the ACM*, 35(4):759–768, 1988.

[DLL62] M. Davis, G. Logemann, and D. Loveland. A machine program for theorem proving. *Communications of the ACM*, 5:394–397, 1962.

[Fri] E. Friedgut. Necessary and sufficient conditions for sharp thresholds of graph properties, and the *k*-sat problem. Preprint, May 1997.

[FS96] A. Frieze and S. Suen. Analysis of two simple heuristics on a random instance of k- SAT. *Journal of Algorithms*, 20(2):312–355, 1996.

[Fu95] Xudong Fu. *On the complexity of proof systems*. PhD thesis, University of Toronto, 1995.

[Goe96] A. Goerdt. A threshold for unsatisfiability. *Journal of Computer and System Sciences*, 53:469–486, 1996.

[GPFW97] J. Gu, P. W. Purdom, J. Franco, and B. J. Wah. Algorithms for the Satisfiability (SAT) Problem: A Survey. In *Satisfiability (SAT) Problem*, DIMACS, pages 19–151. American Mathematical Society, 1997.

[KKK96] L. M. Kirousis, E. Kranakis, and D. Krizanc. Approximating the unsatisfiability threshold of random formulas. In *Proceedings of the Fourth Annual European Symposium on Algorithms*, pages 27–38, Barcelona, Spain, September 1996.

[KS94] S. Kirkpatrick and B. Selman. Critical behavior in the satisfiability of random formulas. *Science*, 264:1297–1301, May 1994.

[Mit95] D. Mitchell. Propositional satisfiability testing. 1995.

[SML96] B. Selman, D. Mitchell, and H. Levesque. Generating hard satisfiability problems. *Artificial Intelligence*, 81:17–29, 1996.