Alcom-FT Technical Report Series
ALCOMFT-TR-02-129

Unsatisfiable Random Formulas are Hard to Certify

Albert Atserias*
Departament de Llenguatges i Sistemes Informàtics
Universitat Politècnica de Catalunya
C/Jordi Girona Salgado, 1-3, Edif. C6.
08034 Barcelona, Spain.
Tel: +34 93 401 69 94
Fax: +34 93 401 70 14
atserias@lsi.upc.es

May 24, 2002

Abstract

We prove that every property of 3CNF formulas that implies unsatisfiability and is expressible in Datalog has asymptotic probability zero when formulas are randomly generated by taking $6n$ non-trivial clauses of exactly three literals uniformly and independently. Our result is a consequence of designing a winning strategy for Duplicator in the existential k-pebble game on the structure that encodes the 3CNF formula and a fixed template structure encoding a satisfiable formula. The winning strategy makes use of certain extension axioms that we introduce and hold almost surely on a random 3CNF formula. An interesting feature of our result is that it brings the fields of Propositional Proof Complexity and Finite Model Theory together. To make this connection more explicit, we show that Duplicator wins the existential pebble game on the structure encoding the Pigeonhole Principle and the template structure above. Moreover, we also prove that there exists a $2k$-Datalog program expressing that an input 3CNF formula has a Resolution refutation of width k. As a consequence to our result and the known size-width relationship in Resolution, we obtain new proofs of the exponential lower bounds for Resolution refutations of random 3CNF formulas and the Pigeonhole Principle.

*Partially supported by CICYT TIC2001-1577-C03-02 and ALCOM-FT IST-99-14186.

1 Introduction

The so-called *phase transition* that certain combinatorial problems exhibit has inspired an alternative line of research within the field of Computational Complexity. The focus is no longer exclusively set to the *worst-case* complexity of the problem. The *average-case* complexity is also considered for certain distributions of interest.

The satisfiability problem for 3CNF formulas, the most popular $\mathbf{NP}$-complete problem, has been intensively studied from this perspective. The distribution of interest in this context is parameterized by the number of variables n and the number of clauses $m = \Delta n$ of the formula. Thus, following [CS88], we let $\mathcal{F}_3^{n,\Delta}$ denote the probability space of random 3CNF formulas with n variables and Δn clauses generated uniformly and independently by selecting 3 variables without replacement from the set of n variables, and negating each variable with probability $1/2$. It turns out that the probability that such a random formula F is satisfiable exhibits a sharp threshold around a critical $\Delta_c \approx 4.2$. Indeed, it has been proven that if $\Delta < 3.26$ then F is almost surely satisfiable [AS00] and if $\Delta > 4.596$ then F is almost surely unsatisfiable [JSV00]. Moreover, experimental results suggest that Δ_c is around 4.2 [SML96], but the existence of such a constant is only a conjecture for the moment [CR92].

This structural analysis of the typical instance of the 3CNF satisfiability problem is complemented by an analysis of the average-case computational complexity of solving it. The first results in this respect were only experimental showing that the running time of the Davis-Putnam-Logemann-Loveland procedure (DPLL) for satisfiability experiments a sharp jump around $\Delta \approx 4.2$, and decays slowly as the ratio Δ increases. These empirical observations are confirmed by two deep theoretical results. On the one hand, Broder, Frieze and Upfal [BFU93] proved that if $\Delta < 1.63$ then a *simple* variant of DPLL running in linear time succeeds in finding a satisfying assignment with high probability. On the other hand, Chvátal and Szeméredi [CS88] proved that if $\Delta > 5.6$ is a constant then every Resolution refutation of an unsatisfiable F almost surely requires size $2^{\Omega(n)}$. Thus, since a DPLL run on F provides a Resolution refutation when F is unsatisfiable no matter which heuristic is used for the splitting rule [BKPS99], this provides a proof that all DPLL-based algorithms behave badly if $\Delta > 5.6$. The results of Broder, Frieze and Upfal were later improved to values of Δ closer to 4.2 (see the nice survey by Achlioptas [Ach01]), and the results of Chvátal and Szeméredi were largely extended to smaller Δ and non-constant Δ [BKPS99, BSW01, BSG01], and to other propositional proof systems such as the Polynomial Calculus [BSI99] and Res(2) [ABE01].

These empirical and theoretical results suggest a stronger form of hardness for satisfiability. Indeed, it seems that for constant values of Δ beyond the critical point, certifying unsatisfiability on a significant fraction of the inputs is computationally hard. A definitive argument in its favor would be proving the *non-existence* of a property of 3CNF formulas that (1) is decidable in polynomial-time, (2) it implies unsatisfiability, and (3) holds with probability bounded away from 0 when $\Delta > 4.2$ is constant. Of course, proving this would imply $\mathbf{P} \neq \mathbf{NP}$ and is therefore out of reach. However, it is possible to obtain some partial unconditional results of this type, and it is clear that they provide some valuable information on our problem.

In this paper we start a descriptive complexity of properties that imply unsatisfiability with non-zero probability. We show the *non-existence* of a property of 3CNF formulas that (1) is expressible in Datalog when 3CNF formulas are encoded as finite relational structures, (2) it implies unsatisfiability, and (3) holds with probability bounded away from 0 when Δ is constant. Datalog is a well-known query language studied in the context of database theory (see Ullman [Ull89]), capable of expressing many interesting problems including Graph Reachability, Non-2-Colorability, and some $\mathbf{P}$-complete problems such as the Monotone Circuit Value Problem and Path Systems among others (see [KV00, Pap85, Ull89]). In a nutshell, Datalog may be viewed as the existential positive fragment of first-order logic augmented with a mechanism of recursion. Let us add that Datalog captures a significant but proper fraction of polynomial-time (see [Ull89] again), and in the presence of a successor relation, it captures polynomial-time exactly [Pap85].

The informed reader will notice that there is an obvious connection between our result and an old result of Blass and Harary [BH79] about random finite graphs. Blass and Harary proved that there is no first-order property of finite graphs that implies Hamiltonicity or Rigidity and has asymptotic probability one under the uniform measure. Both results say that a certain hard computational problem is not implied by an almost sure property that is definable in a significantly expressible logic.

The main ingredient of our proof is the design of a winning strategy for Duplicator in the *existential k-pebble game* between the structure encoding a randomly generated formula, and the structure encoding a fixed template formula that we know is satisfiable. The existential k-pebble game is a powerful combinatorial tool introduced by Kolaitis and Vardi [KV95, KV00] to analyse the expressive power of Datalog with k variables. In fact, the existential k-pebble game captures the expressive power of the stronger logic $\exists\mathcal{L}^k_{\infty\omega}$, the positive existential fragment of infinitary logic with k variables. Thus, our lower bound is much stronger than stated above since what we actually prove is that every property that implies unsatisfiability of formulas with n variables and is (non-uniformly) expressible in the infinitary logic $\exists\mathcal{L}^k_{\infty\omega}$ when $k \leq n/(\ln(n))^2$ must have asymptotic probability 0 when $n \to \infty$ and Δ is constant. Note that we allow the number of variables k to grow with the number of propositional variables n, a parameter that is not allowed to vary in a uniform Datalog program.

The design of the winning strategy for Duplicator relies on the fact that the *incidence graph* of a random 3CNF formula, the bipartite graph that relates clauses with the variables that occur in them, satisfies certain extension axioms that we introduce based on the expander properties of the graph. The use of extension axioms is a recurrent theme in the analysis of pebble games on random structures [Fag76, Lyn80, SS88, KV90]. On the other hand, the expander properties of these graphs have also been used in the context of propositional proof complexity to prove lower bounds in the size of Resolution refutations [BSW01, BSG01]. This establishes an interesting link between the fields of Propositional Proof Complexity and Finite Model Theory. In order to make this connection more explicit, we elaborate on the idea of using the encoding of certain combinatorial principles as propositional tautologies to obtain finite structures on which playing pebble games might be easier than expected. For example, we consider the encoding of the Pigeonhole Principle with n holes as a set of contradictory clauses in the spirit of Cook and Reckhow [CR79], and show that Duplicator wins the existential n-pebble game on the finite structure encoding this formula and the abovementioned template structure. What this result is saying is that the Pigeonhole Principle is a hard example for all Datalog programs that attempt to solve unsatisfiability. This phrasing is inspired by the results of Kolaitis and Vardi relating the Constraint Satisfaction Problems that are solvable by k-Datalog programs, and the finite structures on which Duplicator wins the k-pebble game [KV00].

Finally, we observe that our main inexpressibility result for $\exists\mathcal{L}^k_{\infty\omega}$ is a generalization of the result of Chvátal and Szeméredi. Before we say why, we need to digress shortly to introduce the concept of width in Resolution. In a Resolution refutation, the width is the maximal number of literals in a clause of the refutation. Ben-Sasson and Wigderson proved the following non-trivial result about the width of Resolution: If a 3CNF formula F has a Resolution refutation of size S, then F has a Resolution refutation of width $O\left(\sqrt{n \log S}\right)$, where n is the number of propositional variables. This interesting result relates the width with the size in a form that is suitable to prove size lower bounds. Indeed, if every Resolution refutation of F requires width w, then every Resolution refutation of F requires size $2^{\Omega(w^2/n)}$. What we realize is that there is a tight connection between the width of Resolution, and the number of pebbles needed to win the existential k-pebble game. We show the existence of a Datalog program P with $2k$ variables such that (1) if F has a Resolution refutation of width k, then P evaluates true on the structure encoding F, and (2) if P evaluates true on the structure encoding F, then F has a Resolution refutation of width $2k$. Thus, if Duplicator wins the existential k-pebble game on the structure encoding F and the template structure, then every Resolution refutation of F requires width $k/2$. An immediate corollary of our main result about Duplicator winning the $n/(\ln n)^2$-pebble game on a random 3CNF formula F, and the size-width trade-off of [BSW01], is that every Resolution refutation of F requires size $2^{\Omega(n/(\ln n)^4)}$. The same argument can be used to obtain Haken's [Hak85] exponential lower bound for the Pigeonhole Principle. It should be pointed out that Ben-Sasson and Wigderson prove similar lower bounds by means of the width method in a more direct way; nonetheless, it is quite interesting that non-expressibility results in finite model theory yield lower bounds in proof complexity.

2 Preliminaries

A directed graph $G = (M, E)$ is called *bipartite* if there is a partition of the set of nodes M into two sets U and V such that $E \subseteq U \times V$. A *partial matching* π *of* G is a subset of the edges not sharing an endpoint. If $(u, v) \in \pi$, we write $\pi(u) = v$. Note that this notation is not ambiguous. We say that π *is defined on* u, and we define the *domain* $\mathrm{Dom}(\pi)$ *of* π as the set of nodes of U on which π is defined. We also define the *range* $\mathrm{Ran}(\pi)$ *of* π as $\{\pi(u) : u \in \mathrm{Dom}(\pi)\}$. If $A \subseteq U$, we say that A *is matchable into* V *in* G if there exists a partial matching π of G such that $\mathrm{Dom}(\pi) = A$. If $u \in U$, we let $N_G(u) = \{v \in V : (u, v) \in E\}$, and if $v \in V$, we let $N_G(v) = \{u \in U : (u, v) \in E\}$. The left-degree (resp. right-degree) of G is the maximum size of $N_G(u)$ as u ranges over U (resp. V). If $A \subseteq U$, then $N_G(A) = \bigcup_{u \in A} N_G(u)$. For $\epsilon > 0$, we say that G *is an* (s, ϵ)*-bipartite expander* if for every $A \subseteq U$ of size at most s, the size of $N_G(A)$ is at least $(1 + \epsilon)|A|$. Observe that in an (s, ϵ)-bipartite expander, every subset of U of size at most s is matchable into V. This is true by Hall's Theorem:

Theorem 1 (Hall's Theorem) *If* $G = (U \cup V, E)$ *is a bipartite graph on* U *and* V*, and* $A \subseteq U$*, then* A *is matchable into* V *in* G *if and only if* $|N_G(B)| \geq |B|$ *for every* $B \subseteq A$.

Observe also that an immediate consequence of Hall's Theorem is that if $A \subseteq U$ is minimally non-matchable into V, then $|N_G(A)| < |A|$. We will use this observation in Section 3.

A Datalog program is a finite set of rules of the form

$$t_0 \quad :- \quad t_1, \ldots, t_m$$

where each t_i is an atomic formula $R(x_1, \ldots, x_n)$. The relational symbols that occur in the heads of the rules (left part) are called the *intensional database* predicates (IDBs). The rest of the relational symbols are called the *extensional database* predicates (EDBs). One of the IDBs is called the *goal* of the program, and it might be of arity zero (a propositional letter). A Datalog program is a recursive specification of the IDBs from the EDBs with the least fixed-points semantics (see [Ull89]). In summary, if P is a Datalog program and M is a structure for the vocabulary of the EDBs, the IDBs are given the following semantics: initialze the IDBs to be empty, and repeatedly apply the rules of P whose bodies (right part) are satisfied by M and the current value of the IDBs, until no new facts can be added to the IDBs. We refer the reader to [KV00] for a formal definition of this in terms of the least fixed-points of systems of first-order formulas. Kolaitis and Vardi [KV00] defined k-Datalog as the collection of all Datalog programs in which the body and head of the rules involve at most k distinct variables.

Example 1 [KV00] The class of all finite non-bipartite graphs is definable by a 4-Datalog program. Indeed, the program

$$\begin{aligned} P(x,y) &\ :-\ E(x,y) \\ P(x,y) &\ :-\ P(x,z), E(z,w), E(w,y) \\ Q &\ :-\ P(x,x) \end{aligned}$$

evaluates the goal predicate Q to *true* if and only if E contains an odd cycle, if and only if E is non-bipartite.

Kolaitis and Vardi [KV00] proved that k-Datalog is subsumed by the infinitary logic $\exists\mathcal{L}^k_{\infty\omega}$, that is, the collection of all formulas that have at most k distinct variables and are obtained from the atomic formulas using infinitary disjunction, infinitary conjunction, and existential quantification. Moreover [KV95] also introduced a useful tool to analyse the expressive power of $\exists\mathcal{L}^k_{\infty\omega}$: the existential k-pebble game. The game is played by two players, Spoiler and Duplicator, on two structures A and B over the same relational vocabulary. Each player has k pebbles numbered $\{1,\ldots,k\}$. At each round of the game, Spoiler can make one of two different moves: either he places a free pebble over an element of A, or he removes a pebble from a pebbled element of A. To each move of Spoiler, Duplicator must respond by placing her corresponding pebble over an element of B, or removing her corresponding pebble from B respectively. If Spoiler reaches a round in which the set of pairs of pebbled elements is not a partial homomorphism between A and B, then he wins the game. Otherwise, we say that Duplicator wins the game. Recall that a set of pairs $\{(a_i,b_i)\in A\times B : i=1,\ldots,k\}$ is called a partial homomorphism between A and B if and only if $(b_{i_1},\ldots,b_{i_n})\in R^B$ whenever $(a_{i_1},\ldots,a_{i_n})\in R^A$ and $b_{i_1}=b_{i_2}$ whenever $a_{i_1}=a_{i_2}$. The main result about existential k-pebble games and $\exists\mathcal{L}^k_{\infty\omega}$ is the following [KV00]:

Theorem 2 *Let L be a finite relational vocabulary, let φ be an $\exists\mathcal{L}^k_{\infty\omega}$ sentence for L, and let A, B be two structures for L. If $A\models\varphi$ and Duplicator wins the existential k-pebble game on A and B, then $B\models\varphi$.*

Thus, in order to prove that a class of structures $\mathcal{C}\subseteq\mathcal{D}$ is not definable in k-Datalog over $\mathcal{D}$, it suffices to find $A\in\mathcal{C}$ and $B\in\mathcal{D}-\mathcal{C}$ such that Duplicator wins the existential k-pebble game on A and B.

3 Extension Axioms and the Matching Game

Let $G=(U\cup V,E)$ be a bipartite graph on U and V. The *matching game on G with k fingers* was introduced by Ben-Sasson and Galesi. The game is played by two players: Prover and Disprover. Each player has k fingers numbered $\{1,\ldots,k\}$. In each round of the game, Prover may place a finger over an uncovered node in U or remove a finger from a covered node in U. If Prover places a finger over node $u\in U$, Disprover must place her corresponding finger over an uncovered node in $N_G(u)\subseteq V$. If Prover removes a finger from a node in U, Disprover must remove her corresponding finger from V. The game is over when Disprover is not able to answer to a move of the Prover. In that case, we say that Prover wins the game. If Disprover can make the game go on forever, we say that Disprover wins the game. Notice that at every non-final round, the fingers placed on G determine a partial matching of G. The goal of Disprover is to *maintain* a partial matching forever.

Ben-Sasson and Galesi found a sufficient condition on the graph G that guarantees a win for Disprover. They proved that if G is an (s, ϵ)-bipartite expander (see Section 2 for the definition), then Disprover wins the matching game with $r \leq \epsilon s/(2 + \epsilon)$ fingers. Although our main result would already follow from this, we prefer to give an alternative and self-contained proof of a similar result with the aim of isolating the extension axioms on which the strategy of Disprover relies. The result that we obtain is that if G is an (s, ϵ)-bipartite expander of left-degree at most l, then Disprover wins the matching game with $r \leq \epsilon s/(l + \epsilon)$ fingers. Although our result is weaker than the one of Ben-Sasson and Galesi, our proof is somewhat simpler (Claim 4.6 and 4.7 in [BSG01] are avoided). Moreover, we will consider bipartite graphs whose left-degree is 3, and therefore the weakness of our result is really minor.

Let us isolate an important property of bipartite graphs.

Definition 1 *Let $r \leq s$, let $G = (U \cup V, E)$ be a bipartite graph on U and V, and let $A \subseteq U$ and $B \subseteq V$ be such that $|A| = |B| \leq r$. We say that (G, A, B) satisfies the (r, s)-matching property if the following holds: For every $C \subseteq U - A$ of size $s - |B|$, there exists a partial matching from C to $V - B$ in G.*

The following is the main result of this section.

Theorem 3 *If G is an (s, ϵ)-bipartite expander of left-degree at most l, and $r \leq \epsilon s/(l + \epsilon)$, then Disprover wins the matching game on G with r fingers.*

Proof: The strategy of Disprover is to guarantee that $(G, \mathrm{Dom}(\pi_t), \mathrm{Ran}(\pi_t))$ satisfies the (r, s)-matching property, where π_t is the partial matching of the game at the end of round t. Observe that $(G, \emptyset, \emptyset)$ satisfies the (r, s)-matching property because in an (s, ϵ)-bipartite expander, every subset of U of size s is matchable into V (see Section 2). The following two lemmas will be used to show that Disprover can reply to the moves that Prover makes according to her strategy.

Lemma 1 (Extension Axiom) *If (G, A, B) satisfies the (r, s)-matching property, $u \in U - A$, and $|A| = |B| < r$, then there exists $v \in N_G(u) \cap (V - B)$ such that $(G, A \cup \{u\}, B \cup \{v\})$ satisfies the (r, s)-matching property.*

Proof: Let $\{v_1, \ldots, v_e\} = N_G(u) \cap (V - B)$. Note that $1 \leq e \leq l$ since (G, A, B) satisfies the (r, s)-matching property and therefore every subset of $U - A$ containing u and size $s - |B| > s - r \geq 0$ is matchable into $V - B$. Let $A' = A \cup \{u\}$, and for every $i \in \{1, \ldots, e\}$ let $B^i = B \cup \{v_i\}$. Note that $|A'| = |B^i| \leq r$ since $|A| = |B| < r$. Suppose for contradiction that (G, A', B^i) does not satisfy the (r, s)-matching property for any $i \in \{1, \ldots, e\}$. Let $C^i \subseteq U - A'$ be of size $s - |B^i|$ and non-matchable into $V - B^i$. Let $D^i \subseteq C^i$ be minimally non-matchable into $V - B^i$. Necessarily, $|N_G(D^i) \cap (V - B^i)| < |D^i|$ by Hall's Theorem. The rest of neighbors of D^i can only be in B^i and therefore,

$$|D^i| + |B^i| > N_G(D^i) \geq (1 + \epsilon)|D^i| \tag{1}$$

for every $i \in \{1, \ldots, e\}$. Recall that $|D^i| \leq |C^i| = s - |B^i| \leq s$ and the expansion property of G for the second inequality. We derive a lower bound on $|D^i|$ for some $i \in \{1, \ldots, e\}$. We first claim that $\bigcup_{i=1}^{e} D^i \cup \{u\}$ is not matchable into $V - B$. Indeed, suppose that $\pi \subseteq E$ is such a matching, and let E^i be the image $\pi(D^i)$ of D^i. Then $v_i \in E^i$ since D^i is minimally non-matchable into

$V - B^i$. Therefore, u cannot be matched to any of its neighbors in $V - B$; a contradiction. Now, since $\bigcup_{i=1}^{e} D^i \cup \{u\} \subseteq U - A$ and (G, A, B) satisfies the (r, s)-matching property, we conclude that

$$\left|\bigcup_{i=1}^{e} D^i \cup \{u\}\right| > s - |B|.$$

Therefore, using $|B^i| = |B| + 1$, there must exist some $i \in \{1, \ldots, e\}$ such that

$$|D^i| > (s - |B^i|)/e \geq (s - |B^i|)/l.$$

Plugging this bound into (1) we get $|B^i| > \frac{\epsilon s}{l+\epsilon} \geq r$, a contradiction. □

Lemma 2 (Retraction Axiom) *If (G, A, B) satisfies the (r, s)-matching property and $(u, v) \in E \cap (A \times B)$, then $(G, A - \{u\}, B - \{v\})$ satisfies the (r, s)-matching property.*

Proof: Let $A' = A - \{u\}$ and $B' = B - \{v\}$. Clearly $|A'| = |B'| \leq r$. Let $C \subseteq U - A'$ be of size $s - |B'|$. Suppose first that $u \in C$. Then $C - \{u\} \subseteq U - A$ and has size $s - |B'| - 1 = s - |B|$. Therefore, $C - \{u\}$ is matchable into $V - B$ since (G, A, B) satisfies the (r, s)-matching property. It follows that C is matchable into $V - B'$ by adding the edge (u, v) to that matching. Suppose next that $u \notin C$. For every $w \in C$, the set $C - \{w\} \subseteq U - A$ is matchable into $V - B \subseteq V - B'$ since (G, A, B) satisfies the (r, s)-matching property. It follows that if C is not matchable into $V - B'$, then it is minimally non-matchable, and so $|N_G(C) \cap (V - B')| < |C|$ by Hall's Theorem. The rest of neighbors of C can only be in B' and therefore

$$|C| + |B'| > |N_G(C)| \geq (1 + \epsilon)|C|.$$

Recall that $|C| = s - |B'| \leq s$ and the expansion property of G for the second inequality. We conclude that $|B'| > \frac{\epsilon s}{1+\epsilon} \geq \frac{\epsilon s}{l+\epsilon} \geq r$, a contradiction. □

We are ready to complete the proof of the theorem. Suppose that $(G, \mathrm{Dom}(\pi_t), \mathrm{Ran}(\pi_t))$ satisfies the (r, s)-matching property. Consider the move of Prover at time $t + 1$. If Prover places a finger over an uncovered node $u \in U - \mathrm{Dom}(\pi_t)$, Disprover will answer by some $v \in N_G(u) \cap (V - \mathrm{Ran}(\pi_t))$ such that $(G, \mathrm{Dom}(\pi_t) \cup \{u\}, \mathrm{Ran}(\pi_t) \cup \{v\})$ satisfies the (r, s)-matching property. Such a v exists by Lemma 1 because $|\mathrm{Dom}(\pi_t)| = |\mathrm{Ran}(\pi_t)| < r$. If Prover removes a finger from a node $u \in \mathrm{Dom}(\pi_t)$, Disprover simply removes the corresponding finger from $v = \pi_t(u)$, and $(G, \mathrm{Dom}(\pi_t) - \{u\}, \mathrm{Ran}(\pi_t) - \{v\})$ will satisfy the (r, s)-matching property by Lemma 2. □

4 On Sufficient Conditions for Unsatisfiability

Let $v_1, v_2, \ldots$ be an infinite supply of propositional variables. A literal is a variable or the negation of a variable. A k-clause is a set of k literals (repeated and complementary literals are allowed). A k-CNF formula is a set of k-clauses. Let $F = C_1 \wedge \ldots \wedge C_m$ be a k-CNF formula on the variables $v_1, \ldots, v_n$. We encode F as a finite structure $(M, R_0, \ldots, R_k)$ with $k + 1$ relations of arity k. The universe M is the set of variables $\{v_1, \ldots, v_n\}$. For each $s \in \{0, \ldots, k\}$, the relation R_s encodes the set of clauses of F with exactly s negated variables. More precisely, R_s consists of all k-tuples of the form

$$(v_{i_1}, \ldots, v_{i_s}, v_{j_1}, \ldots, v_{j_{k-s}}) \in \{v_1, \ldots, v_n\}^k$$

such that $\{\neg v_{i_1}, \ldots, \neg v_{i_s}, v_{j_1}, \ldots, v_{j_{k-s}}\}$ is a clause of F. The encoding of F by such a finite structure will be denoted by $M(F)$.

Example 2 Consider the 3CNF formula $F = (v_1 \vee \neg v_2 \vee v_3) \wedge (\neg v_2 \vee \neg v_3 \vee v_4) \wedge (v_1 \vee \neg v_3 \vee \neg v_4)$ over the variables $v_1, \ldots, v_4$. Recall that, strictly speaking, the clauses of F are sets of literals. Thus, $C_1 = \{\neg v_2, v_1, v_3\}$, $C_2 = \{\neg v_2, \neg v_3, v_4\}$ and $C_3 = \{\neg v_3, \neg v_4, v_1\}$. The structure encoding F is $M(F) = (M, R_0, R_1, R_2, R_3)$ where $M = \{v_1, v_2, v_3, v_4\}$, $R_0 = \emptyset$, $R_1 = \{(2,1,3), (2,3,1)\}$, $R_2 = \{(2,3,4), (3,2,4), (3,4,1), (4,3,1)\}$ and $R_3 = \emptyset$.

Example 3 We define a template structure $T = (\{0,1\}, R_0^T, \ldots, R_k^T)$ where

$$R_s^T = \{(a_1, \ldots, a_s, b_1, \ldots, b_{k-s}) \in \{0,1\}^k : \neg a_1 \vee \ldots \vee \neg a_s \vee b_1 \vee \ldots \vee b_{k-s} = 1\}$$

for every $s \in \{0, \ldots, k\}$. Observe that T is isomorphic to the encoding of a k-CNF formula on the variables v_0 and v_1. Moreover, it is not hard to see that this formula is satisfiable by setting v_1 to 1 and v_0 to 0.

Let $F = C_1 \wedge \ldots \wedge C_m$ be a 3CNF formula on the variables $v_1, \ldots, v_n$. We define the *incidence graph $G(F)$ of F* as follows: $G(F)$ is a bipartite graph on the sets of nodes $\{C_1, \ldots, C_m\}$ and $\{v_1, \ldots, v_n\}$, and (C_i, v_j) is an edge in $G(F)$ if and only if $v_j \in Var(C_i)$. Here $Var(C_i)$ is used to denote the set of variables that occur in C_i. Notice that the left-degree of $G(F)$ is at most 3.

Let F be a 3CNF formula, let V be a set of variables from F, and let $\mathcal{C} \subseteq F$ be the subset of the clauses of F that mention some variable from V. In symbols,

$$\mathcal{C} = \{C \in F : Var(C) \cap V \neq \emptyset\}.$$

Let π be a partial matching of $G(F)$ such that $\mathrm{Dom}(\pi) = \mathcal{C}$. We define the *partial truth assignment ρ corresponding to π and V* as follows: For every $x \notin V$, the partial truth assignment ρ is undefined on x. If $x \in V$ and $\pi(C) = x$ for some $C \in \mathcal{C}$, then $\rho(x) = 0$ if x occurs both positively and negatively in C, $\rho(x) = 1$ if x occurs positively in C only, and $\rho(x) = 0$ if x occurs negatively in C only. Finally, if $x \in V$ but $x \notin \mathrm{Ran}(\pi)$, then $\rho(x) = 0$. Note that ρ is well-defined because π is a partial matching. In the following, we say that a partial truth assignment ρ *falsifies a clause C* if ρ sets all literals of C to 0.

Lemma 3 *Let F be a 3CNF formula and let ρ be the partial truth assignment corresponding to some partial matching π of $G(F)$ and some set of variables V. Then ρ does not falsify any clause from F.*

Proof: Let $\mathcal{C}$ be the set of clauses from F that mention some variable in V. Let C be an arbitrary clause from F. If $C \notin \mathcal{C}$, then every variable of C is left undefined by ρ and C cannot be falsified. If $C \in \mathcal{C}$, then $\pi(C)$ occurs in C and $\rho(\pi(C))$ is set to satisfy C. □

For the following result, recall the definition of (s, ϵ)-bipartite expander in Section 2 and the definition of the template structure T in Example 3.

Theorem 4 *Let F be a 3CNF formula such that $G(F)$ has right-degree at most d. If Disprover wins the matching game with r fingers on $G(F)$, then Duplicator wins the existential $\lfloor r/d \rfloor$-pebble game on $M(F)$ and T.*

Proof: Let $G = G(F)$ and $r' = \lfloor r/d \rfloor$. Thus, G is a bipartite graph between the clauses and the variables of F. Duplicator will simulate a play of the matching game on G on the side trying to satisfy the following condition:

Clause C has a finger on in it in the matching game at time t if and only if some variable $x \in Var(C)$ has a pebble on it in the existential r'-pebble game at time t.

In other words, if V_t is the set of pebbled variables in the existential r'-pebble game at the end of round t, and $\mathcal{C}_t$ is the set of clauses of F that mention some variable in V_t, then $\mathcal{C}_t$ is the set of fingered clauses in the matching game at the end of round t. Since at most r' variables are pebbled simultaneously in the existential r'-pebble game, and since the right degree of G is at most d, at most $r'd = r$ fingers will be required. Duplicator will make use of the winning strategy of Disprover in the matching game on G with r fingers. In the following, let π_t be the partial matching in the matching game at the end of round t. Note that $\pi_0 = \emptyset$ and $\mathrm{Dom}(\pi_t) = \mathcal{C}_t$. We consider the two possible moves of Spoiler at time $t+1$ separately.

If Spoiler puts a pebble on variable x, Duplicator acts as follows. Let $\mathcal{D} = \{C_1, \ldots, C_l\}$ be the set of clauses of F on which variable x occurs. Duplicator considers each clause of $\mathcal{D}$ one at a time, starting by C_1. If C_1 does not have a finger on it in the matching game at time t, Duplicator simulates the winning strategy of Disprover as if Prover had placed a finger over C_1, and proceeds to C_2. Otherwise, she proceeds to C_2 directly. This procedure is repeated until C_l is considered and treated. Let π_{t+1} be the partial matching in the matching game at that point. Then Duplicator replies in the existential r'-pebble game as follows: If some C_i exists such that $\pi_{t+1}(C_i) = x$, Duplicator answers 0 if x occurs both positively and negatively in C_i, answers 1 if x occurs positively in C_i only, and answers 0 if x occurs negatively in C_i only. Otherwise, Duplicator answers 0. Note for the record that Duplicator is answering consistently with the partial truth assignment corresponding to π_{t+1} and V_{t+1}, the partial matching and the set of pebbled variables in the existential r'-pebble game at time $t+1$.

If Spoiler removes a pebble from variable x, the answer of Duplicator is clear: she removes the corresponding pebble from T. However, she will need to do some bookeeping in the matching game: Let $\mathcal{D} = \{C_1, \ldots, C_l\}$ be the set of clauses of F on which variable x occurs. Duplicator considers each clause of $\mathcal{D}$ one at a time, starting by C_1. If some other variable $y \in Var(C_1) - \{x\}$ has a pebble on it in the existential r'-pebble game at time t, Duplicator proceeds to C_2. If there is no $y \in Var(C_1) - \{x\}$ as above, Duplicator simulates the matching game as if Prover had removed his finger from C_1. Then Duplicator proceeds to C_2. This procedure is repeated until C_l is considered and treated. Let π_{t+1} be the partial matching in the matching game at that point.

We need to argue two points: (1) that Duplicator can follow this strategy, and (2) that if she does, she wins the existential r'-pebble game on $M(F)$ and T. For (1), recall that each variable occurs in at most d clauses, and therefore, since at most $r' = \lfloor r/d \rfloor$ variables are pebbled in the existential r'-pebble game, at most r fingers are used in the matching game. It follows that Duplicator can simulate the winning strategy of Disprover in the matching game. For (2), suppose that Spoiler has not won at time t; we show that he does not win at time $t+1$. When Spoiler removes a pebble, Duplicator cannot loose since she removes the corresponding pebble from T. When Spoiler puts a new pebble, Duplicator answers according to the partial truth assignment corresponding to the partial matching π_{t+1} and the set of pebbled variables V_{t+1}, and this cannot falsify a clause from F by Lemma 3. This completes the proof of the Theorem. $\Box$

Let 3CNF$_n$ be the class of all structures of the form $M(F)$ where F is a 3CNF formula on at most n variables. Let 3SAT$_n \subseteq$ 3CNF$_n$ be the class of all structures of the form $M(F)$ where F is a satisfiable 3CNF formula on at most n variables. Similarly, let 3UNSAT$_n \subseteq$ 3CNF$_n$ be the class of all structures of the form $M(F)$ where F is an unsatisfiable 3CNF formula on at most n variables. Recall the definition of the distribution $\mathcal{F}_3^{n,\Delta}$ from the introduction.

Theorem 5 *For all constants $\delta > 0$, $\Delta > 0$, and for all sufficiently large n, if $C_n \subseteq$ 3CNF$_n$ is such that*

(i) $C_n \subseteq$ 3UNSAT$_n$ and

(ii) $\Pr[M(F) \in C_n] \geq \delta$ when F is drawn from $\mathcal{F}_3^{n,\Delta}$,

then C_n is not definable in $\exists\mathcal{L}^k_{\infty\omega}$ on 3CNF$_n$ for $k \leq n/(\ln(n))^2$.

Proof: Fix $\delta > 0$, $\Delta > 0$, and let n be large. Let $k \leq n/(\ln(n))^2$. Suppose for contradiction that C_n satisfies (i) and (ii), and φ is an $\exists\mathcal{L}^k_{\infty\omega}$ sentence defining C_n on 3CNF$_n$. We will show that there exists a 3CNF formula F such that $M(F) \in C_n$ and $G(F)$ is an $(n/\ln(n), 1/4)$-bipartite expander of right-degree at most $d = \ln(n)/13$ and left-degree at most 3. This will be enough because then: (1) $M(F) \models \varphi$ and (2) Duplicator wins the existential k-pebble game on $M(F)$ and T by Theorem 3 and Theorem 4. This means that $T \models \varphi$ by Theorem 2, so that $T \in C_n \subseteq$ 3UNSAT$_n$; a contradiction since T is isomorphic to a satisfiable 3CNF formula on at most n variables (in fact, on two variables).

Let us prove the existence of F as above by the probabilistic method. First we bound the probability that the right degree of $G(F)$ is bigger than d. Let $d_R(G(F))$ denote the right-degree of the bipartite graph $G(F)$.

Lemma 4 *Let F be drawn from $\mathcal{F}_3^{n,\Delta}$. Then,*

$$\Pr\left[d_R(G(F)) \geq d\right] \leq n\left(\frac{3e\Delta}{d}\right)^d.$$

Proof: Note that a variable v_i occurs in a random clause with probability $\binom{n-1}{2}\binom{n}{3}^{-1} = 3/n$. For every $i \in \{1,\ldots,n\}$, let X_i be the random variable that counts the number of clauses of F in which variable v_i occurs. By the above, the expectation of X_i is $\Delta n \cdot 3/n = 3\Delta$. More important,

$$\Pr[X_i \geq d] \leq \binom{\Delta n}{d}\left(\frac{3}{n}\right)^d \leq \left(\frac{\Delta n e}{d}\right)^d\left(\frac{3}{n}\right)^d = \left(\frac{3e\Delta}{d}\right)^d.$$

Here we used the well-known inequality $\binom{n}{k} \leq (ne/k)^k$ that is derivable using the fact that $k! \geq (k/e)^k$. Now, let X be the random variable that counts the number of $i \in \{1,\ldots,n\}$ such that $X_i \geq d$. The expectation of X is at most $n(3e\Delta/d)^d$. Therefore, the probability that $X > 0$ is at most $n(3e\Delta/d)^d$ by Markov's inequality. □

Next we bound the probability that $G(F)$ is not an (s,ϵ)-bipartite expander.

Lemma 5 *Let $s > 0$ and $\epsilon > 0$ be such that $(1+\epsilon)s \leq n$ and let F be drawn from $\mathcal{F}_3^{n,\Delta}$. Then,*

$$\Pr\left[G(F) \textit{ not } (s,\epsilon)\textit{-bipartite expander}\right] \leq \frac{C(1-C^s)}{1-C},$$

where $C = \Delta e^{2+\epsilon}(1+\epsilon)^{2-\epsilon}(s/n)^{1-\epsilon}$.

Proof: Write $G = G(F) = (U_1 \cup U_2, E)$. Let $\gamma = 1+\epsilon$. We bound the probability that for a fixed set $U \subseteq U_1$ of size $i \leq s$ and a fixed set $V \subseteq U_2$ of size γi it holds that $N_G(U) \subseteq V$. For a fixed $u \in U$, the probability that $N_G(u) \subseteq V$ is bounded by $\binom{\gamma i}{3}\binom{n}{3}^{-1}$. Since each clause is drawn independently of the rest, the probability that $N_G(U) \subseteq V$ is bounded by

$$\left[\frac{\binom{\gamma i}{3}}{\binom{n}{3}}\right]^i \leq \left(\frac{\gamma i}{n}\right)^{3i}.$$

Here we used the fact that $\binom{n}{k} / \binom{m}{k} \leq (n/m)^k$ if $n \leq m$. Therefore, the probability that G is not an (s,ϵ)-bipartite expander is bounded by

$$\sum_{i=1}^{s} \binom{\Delta n}{i}\binom{n}{\gamma i}\left(\frac{\gamma i}{n}\right)^{3i} \leq \sum_{i=1}^{s}\left(\frac{\Delta n e}{i}\right)^{i}\left(\frac{ne}{\gamma i}\right)^{\gamma i}\left(\frac{\gamma i}{n}\right)^{3i} = \sum_{i=1}^{s}\left[\Delta e^{2+\epsilon}(1+\epsilon)^{2-\epsilon}\left(\frac{i}{n}\right)^{1-\epsilon}\right]^{i}.$$

The largest term into brackets is achieved when $i = s$. Thus, we can bound the sum by

$$\sum_{i=1}^{s} C^i \leq \frac{C(1-C^s)}{1-C}$$

as was to be proved. □

Finally, we bound the probability that the right-degree of $G(F)$ is bigger than d, or $G(F)$ is not an (s,ϵ)-expander, or $M(F)$ does not belong to C_n by

$$n\left(\frac{3e\Delta}{d}\right)^d + \frac{C(1-C^s)}{1-C} + 1 - \delta. \tag{2}$$

From our choice of $d = \ln(n)/13$, $s = n/\ln(n)$, and $\epsilon = 1/4$, we immediately see that (2) is strictly smaller than 1 for sufficiently large n (recall that $\delta > 0$ and $\Delta > 0$ are constants). Thus, some F must exist for which $G(F)$ is an (s,ϵ)-expander of right-degree at most d, and $M(F)$ belongs to C_n. This completes the proof of the Theorem. □

Let us recall that every query expressible in Datalog is also expressible in $\exists\mathcal{L}^{\omega}_{\infty\omega}$. Thus, an immediate (much weaker but easier to state) corollary of Theorem 5 is the following:

Corollary 1 *There is no Datalog property of 3CNF formulas that implies unsatisfiability and has non-zero asymptotic probability in* $\mathcal{F}_3^{n,6}$.

5 Easier Games

The aim of this section is to show that certain combinatorial statements such as the Pigeonhole Principle, when encoded as contradictory sets of clauses in the spirit of Cook and Reckhow [CR79], may yield structures on which playing pebble games is easier than expected.

The Pigeonhole Principle states that there is no one-to-one mapping from a set of $n+1$ pigeons into a set of n holes. This statement may be expressed in propositional logic as follows: For $i \in \{1,\ldots,n+1\}$ and $j \in \{1,\ldots,n\}$, let $p_{i,j}$ be a propositional variable meaning that pigeon i is sitting in hole j. The clause $P_i = p_{i,1} \vee \ldots \vee p_{i,n}$ says that pigeon i sits in some hole. The clause $H_k^{i,j} = \neg p_{i,k} \vee \neg p_{j,k}$ for $i \neq j$ says that pigeons i and j cannot sit in the same hole k. The conjunction PHP_n^{n+1} of all the clauses P_i and $H_k^{i,j}$ is an unsatisfiable CNF since it expresses the negation of the Pigeonhole Principle. Note that PHP^{n+1}_n is not a 3CNF formula since the clauses P_i involve n literals. However, it is easy to turn it into an equivalent 3CNF as follows: For every $i \in \{1, \ldots, n+1\}$ and $j \in \{0, \ldots, n\}$, let $y_{i,j}$ be a new propostional variable. Then EP_i is the following 3CNF formula: $\neg y_{i,0} \wedge \bigwedge_{j=1}^{n}(y_{i,j-1} \vee p_{i,j} \vee \neg y_{i,j}) \wedge y_{i,n}$. Finally, the 3CNF formula $EPHP^{n+1}_n$ expressing the negation of the Pigeonhole Principle is the conjunction of all clauses EP_i and all clauses $H^{i,j}_k$. Here is our promised result about pebble games and the Pigeonhole Principle:

Theorem 6 *Duplicator wins the existential n-pebble game on $M(EPHP^{n+1}_n)$ and T.*

Proof: Consider the following strategy for Duplicator. She will keep track of a partial matching π_t in the complete bipartite graph K^{n+1}_n with the following property:

> Pigeon $i \in \{1, \ldots, n+1\}$ belongs to $\mathrm{Dom}(\pi_t)$ if and only if some variable of pigeon i is pebbled at the end of round t.

Of course, the variables of pigeon i are $p_{i,1}, \ldots, p_{i,n}$ and $y_{i,0}, \ldots, y_{i,n}$. Notice that if there are p pebbles on the board at the end of round t, then $|\mathrm{Dom}(\pi_t)| = |\mathrm{Ran}(\pi_t)| \leq p$. Initially $\pi_0 = \emptyset$. Let us consider the possible moves of Spoiler at time $t+1$ separately:

If Spoiler places a pebble over a variable $p_{i,j}$, Duplicator acts as follows: If $i \in \mathrm{Dom}(\pi_t)$, Duplicator answers 1 if $j = \pi_t(i)$ and 0 otherwise. In that case, $\pi_{t+1} = \pi_t$. If $i \notin \mathrm{Dom}(\pi_t)$, Duplicator finds some $k \in \{1, \ldots, n\} - \mathrm{Ran}(\pi_t)$ and sets $\pi_{t+1} = \pi_t \cup \{(i,k)\}$. Note that such a k exists since at most $n-1$ pebbles are placed on the board at the end of round t, and therefore $|\mathrm{Dom}(\pi_t)| = |\mathrm{Ran}(\pi_t)| \leq n-1$. Then she answers 1 if $j = k$ and 0 otherwise.

If Spoiler places a pebble over a variable $y_{i,j}$, Duplicator acts as follows: If $i \in \mathrm{Dom}(\pi_t)$, Duplicator answers 1 if $j \geq \pi_t(i)$ and 0 otherwise. In that case, $\pi_{t+1} = \pi_t$. If $i \notin \mathrm{Dom}(\pi_t)$, Duplicator finds some $k \in \{1, \ldots, n\} - \mathrm{Ran}(\pi_t)$ and sets $\pi_{t+1} = \pi_t \cup \{(i,k)\}$. Such a k exists for the same reason as above. Then she answers 1 if $j \geq k$ and 0 otherwise.

If Spoiler removes a pebble from a variable, the answer of Duplicator is clear: she removes the corresponding pebble in T. However, she will need to do some bookeeping in the partial matching π_t. Suppose that the variable from which Spoiler removed the pebble belongs to pigeon i. If no other variable of pigeon i is pebbled at time t, Duplicator sets $\pi_{t+1} = \pi_t - \{(i, \pi_t(i))\}$. Otherwise, Duplicator sets $\pi_{t+1} = \pi_t$.

It is not hard to see that Duplicator wins the existential n-pebble game when playing according to this strategy since a clause is never falsified. $\Box$

We could have done the same with a form of the Pigeonhole Principle that is based on a bipartite graph. Let U and V be disjoint sets of sizes $n+1$ and n respectively. Let $G = (U \cup V, E)$ be a bipartite graph on U and V. Obviously, there is no matching from U into V. This is expressed by the conjunction of the following set of clauses on the variables $\{x_e : e \in E\}$. For every $u \in U$, let $P_u = \bigvee_{(u,v) \in E} x_{u,v}$, and for every $(u,w), (v,w) \in E$, $u \neq v$, let $H^{u,v}_w = \neg x_{u,w} \vee \neg x_{v,w}$. We write G-PHP for the conjunction of these clauses. Observe that if $G = K^{n+1}_n$, then G-PHP coincides with PHP^{n+1}_n. Observe also that if G has left-degree at most k, then G-PHP is a k-CNF.

Theorem 7 *If $G = (U \cup V, E)$ is an (s, ϵ)-expander of left-degree at most l, and $r \leq \epsilon s/(l+\epsilon)$, then Duplicator wins the existential r-pebble game on $M(G$-$PHP)$ and T.*

Proof idea: Design a strategy for Duplicator to keep track of a partial matching in G with the property that a pigeon $u \in U$ is in the domain of the partial matching at time t if and only if some variable of pigeon u is pebbled at time t. $\Box$

6 Connections with Proof Complexity and Resolution

Resolution is probably the most popular proof system for propositional logic. There is only one rule: from clauses $C \cup \{x\}$ and $D \cup \{\neg x\}$ derive $C \cup D$. A Resolution refutation of a CNF formula F is a sequence of clauses ending with the empty clause $\{\}$, each of which is either a clause of F, or follows from two previous clauses in the sequence by the Resolution rule. The length of a Resolution refutation is the length of the sequence of clauses. The width of a Resolution refutation is the maximum number of distinct literals in a clause of the refutation.

Theorem 8 *There exists a $2k$-Datalog sentence φ such that for every 3CNF formula F, the following holds:*

(i) If F has a Resolution refutation of width k, then $M(F) \models \varphi$.

(ii) If $M(F) \models \varphi$, then F has a Resolution refutation of width $2k$.

Proof: Consider the following set of $2k$-Datalog rules. For every w and i such that $0 \leq i \leq w \leq 2k$, let S_i^w be a new predicate symbol of arity w.

$$S_i^3(x_1, x_2, x_3) \quad :- \quad R_i(x_1, x_2, x_3) \tag{3}$$

The meaning of (3) is that every clause of F is derivable. Add also the following rules:

$$S_i^w(x_1, \ldots, x_p/x_q, \ldots, x_q/x_p, \ldots, x_i, y_1, \ldots, y_{w-i}) \quad :- \quad S_i^w(x_1, \ldots, x_i, y_1, \ldots, y_{w-i}) \tag{4}$$
$$S_i^w(x_1, \ldots, x_i, y_1, \ldots, y_{p'}/y_{q'}, \ldots, y_{q'}/y_{p'}, \ldots, y_{w-i}) \quad :- \quad S_i^w(x_1, \ldots, x_i, y_1, \ldots, y_{w-i}) \tag{5}$$

where $1 \leq p < q \leq i$ in (4), $1 \leq p' < q' \leq w - i$ in (5). The meaning of (4) is that if the clause $\{\neg x_1, \ldots, \neg x_i, y_1, \ldots, y_{w-i}\}$ is a derivable, then any result of permuting two x-literals in it is also derivable. Similarly for (5).

$$S_i^w(x, \ldots, x, x, x_1, \ldots, x_a, y_1, \ldots, y_{w-i}) \quad :- \quad S_i^w(x, \ldots, x, x_1, x_1, \ldots, x_a, y_1, \ldots, y_{w-i}) \tag{6}$$
$$S_i^w(x_1, \ldots, x_i, y_1, \ldots, y_b, y, y, \ldots, y) \quad :- \quad S_i^w(x_1, \ldots, x_i, y_1, \ldots, y_b, y_b, y, \ldots, y) \tag{7}$$
$$S_i^w(x_1, \ldots, x_i, y_1, \ldots, y_{w-i-1}, y_{w-i-1}) \quad :- \quad S_{i+1}^w(x_1, x_1, \ldots, x_i, y_1, \ldots, y_{w-i-1}) \tag{8}$$
$$S_{i+1}^w(x_1, x_1, \ldots, x_i, y_1, \ldots, y_{w-i-1}) \quad :- \quad S_i^w(x_1, \ldots, x_i, y_1, \ldots, y_{w-i-1}, y_{w-i-1}) \tag{9}$$

where $0 < a < i - 1$ in (6), $0 < b < w - i - 1$ in (7), and $0 < i < w - 1$ in (8) and (9). The meaning of rule (6) is that if $\{\neg x, \ldots, \neg x, \neg x_1, \neg x_1, \ldots, \neg x_a, y_1, \ldots, y_{w-i}\}$ is derivable, so is the result of contracting one $\neg x_1$ and adding one $\neg x$. Similarly for (7). The meaning of (8) is that if $\{\neg x_1, \neg x_1, \ldots, \neg x_i, y_1, \ldots, y_{w-i-1}\}$ is derivable, so is the result of contracting one $\neg x_1$ and adding one y_{w-i-1}. Similarly for (9). We introduce two rules that, together with (4), . . . ,(9) can be used to eliminate repeated literals.

$$S_i^w(x_1, \ldots, x_i, y_1, \ldots, y_{w-i}) \quad :- \quad S_i^{w+p}(x_1, \ldots, x_i, y_1, \ldots, y_{w-i}, y_{w-i}, \ldots, y_{w-i}) \tag{10}$$
$$S_i^w(x_1, \ldots, x_i, y_1, \ldots, y_{w-i}) \quad :- \quad S_{i+p}^{w+p}(x_1, \ldots, x_1, x_1, \ldots, x_i, y_1, \ldots, y_{w-i}), \tag{11}$$

as long as $w + p \leq 2k$, $0 \leq i < w$ in (10) and $0 < i \leq w$ in (11). Next we introduce the rule that simulates the Resolution rule:

$$S_{r_1+r_2-1}^{w_1+w_2-2}(x_2, \ldots, x_{r_1}, x'_1, \ldots, x'_{r_2}, y_1, \ldots, y_{w_1-r_1}, y'_1, \ldots, y'_{w_2-r_2-1}) :- \tag{12}$$
$$:- \quad S_{r_1}^{w_1}(z, x_2, \ldots, x_{r_1}, y_1, \ldots, y_{w_1-r_1}), S_{r_2}^{w_2}(x'_1, \ldots, x'_{r_2}, y'_1, \ldots, y'_{w_2-r_2-1}, z), \tag{13}$$

where $w_1 + w_2 - 2 \le 2k$, $0 < r_1 \le w_1$ and $0 \le r_2 < w_2$. The meaning of (13) is that if the clauses $\{\neg z, \neg x_2, \ldots, \neg x_{r_1}, y_1, \ldots, y_{w_1 - r_1}\}$ and $\{\neg x'_1, \ldots, \neg x'_{r_2}, y'_1, \ldots, y'_{w_2 - r_2 - 1}, z\}$ have been derived, then the resolvent clause is also derivable by an application of the Resolution rule. Note that the resulting clause may have repeated literals. Note also that we do not insist that variable z appears only once in each clause in order to apply the Resolution rule. The soundness of the rule is not lost and we will make use of this fact later. Finally, add a new predicate symbol G and the rule

$$G \quad :- \quad S^1_0(x), S^1_k(x). \tag{14}$$

The meaning of (14) is that if $\{x\}$ and $\{\neg x\}$ are derivable, then F is refutable.

Let φ be the conjunction of all these Datalog rules. It is a routine task (but somewhat tedious) to check that if F has a Resolution refutation of width k then $M(F) \models \varphi$. Similarly, if $M(F) \models \varphi$ then F has a Resolution refutation of width $2k$. The proof follows.

We first prove that if F has a Resolution refutation of width k, then $M(F) \models \varphi$. Indeed, suppose that $C_1, C_2, \ldots, C_m$ is a Resolution refutation of F of width k. We may assume without loss of generality that $C_{m-2} = \{x\}$, $C_{m-1} = \{\neg x\}$ and $C_m = \{\}$. For $j \in \{1, \ldots, m-1\}$, let us write $C_j = \{\neg x_{j,1}, \ldots, \neg x_{j,r_j}, y_{j,1}, \ldots, y_{j,s_j}\}$ where all displayed variables are different and $r_j + s_j \le k$. Let $w_j = r_j + s_j$ be the width of C_j. We will show, by induction on $j \in \{1, \ldots, m-1\}$, that

$$S^{w_j}_{r_j}(x_{j,1}, \ldots, x_{j,r_j}, y_{j,1}, \ldots, y_{j,s_j})$$

is derivable from $M(F)$ and the Datalog rules. Fix $j \ge 1$ and suppose that the claim holds for all smaller values of j. If C_j is a clause of F, the conclusion is immediate from rule (3), and rules (4) $, \ldots,$(11). Suppose then that C_j follows from C_h and C_l by the Resolution rule. Here $h < l < j$. By induction hypothesis, we know that

$$S^{w_h}_{r_h}(x_{h,1}, \ldots, x_{h,r_h}, y_{h,1}, \ldots, y_{h,s_h}) \tag{15}$$

$$S^{w_l}_{r_l}(x_{l,1}, \ldots, x_{l,r_l}, y_{l,1}, \ldots, y_{l,s_l}) \tag{16}$$

are derivable. Since C_h and C_l are resolved together to obtain C_j, some variable must appear negatively in C_h and positively in C_l, or conversely. Without loss of generality, let us assume that the former holds. Using the rules (4) and (5), we may also assume that $x_{h,1} = y_{l,s_l}$ is the cut variable z. Now let us apply the rule (13) to obtain

$$S^{w_h + w_l - 2}_{r_h + r_l - 1}(x_{h,2}, \ldots, x_{h,r_h}, x_{l,1}, \ldots, x_{l,r_l}, y_{h,1}, \ldots, y_{h,s_h}, y_{l,1}, \ldots, y_{l,s_l - 1}).$$

Then apply the rules (4), $\ldots$,(9) to obtain

$$S^{w_h + w_l - 2}_{r_j}(x_{j,1}, \ldots, x_{j,r_j}, y_{j,1}, \ldots, y_{j,s_j}, y_{j,s_j}, \ldots, y_{j,s_j}).$$

or

$$S^{w_h + w_l - 2}_{w_h + w_l - 2}(x_{j,1}, \ldots, x_{j,1}, x_{j,1}, \ldots, x_{j,r_j})$$

depending on whether $s_j > 0$ or $s_j = 0$ respectively. Finally, apply the rule (10) or (11) to obtain

$$S^{w_j}_{r_j}(x_{j,1}, \ldots, x_{j,r_j}, y_{j,1}, \ldots, y_{j,s_j}).$$

We have proved that $S^1_0(x)$ and $S^1_1(x)$ are derivable. Thus, G is derivable and so $M(F) \models \varphi$.

Let us prove the reverse direction. Namely, that if $M(F) \models \varphi$, then F has a Resolution refutation of width $2k$. In order to prove that, we will need to introduce a mild extension of Resolution as an intermediary step. The new proof system incorporates the so-called weakening rule: from clause C, derive $C \cup D$ where D is an arbitrary clause. It can be proved by induction on the length of the proof that weakenings may be eliminated without increasing the width, and therefore allowing them cannot hurt. Next we prove that if

$$S_i^w(x_1, \ldots, x_i, y_1, \ldots, y_{w-i})$$

holds, then the clause

$$\{\neg x_1, \ldots, \neg x_i, y_1, \ldots, y_{w-i}\} \tag{17}$$

has a Resolution derivation with weakening from F in width $2k$. The proof is by induction on the stage s at which the tuple $(x_1, \ldots, x_i, y_1, \ldots, y_{w-i})$ entered in the least fixed-point of the Datalog rules. Let $s > 0$ and suppose that the claim holds for all smaller values of s. If the tuple entered following rule (3), the claim is obvious since then (17) is an initial clause of F. If the tuple entered following one of the rules (4), $\ldots$,(11), the claim is also clear since clauses are sets of literals. Finally, if the tuple entered following the rule (13), we proceed as follows: Let us rename $x_1, \ldots, x_i, y_1, \ldots, y_{w-i}$ in the form of the conclusion of rule (13); that is

$$(x'_2, \ldots, x'_{r_1}, x''_1, \ldots, x''_{r_2}, y'_1, \ldots, y'_{w_1-r_1}, y''_1, \ldots, y''_{w_2-r_2-1}), \tag{18}$$

where both

$$\begin{array}{l} S_{r_1}^{w_1}(z, x'_2, \ldots, x'_{r_1}, y'_1, \ldots, y'_{w_1-r_1}) \\ S_{r_2}^{w_2}(x''_1, \ldots, x''_{r_2}, y''_1, \ldots, y''_{w_2-r_2-1}, z) \end{array}$$

hold and have entered in a previous stage. If z does not occur in (18), then (17) is simply the result of resolving $\{\neg z, \neg x'_2, \ldots, \neg x'_{r_1}, y'_1, \ldots, y'_{w_1-r_1}\}$ and $\{\neg x''_1, \ldots, \neg x''_{r_2}, y''_1, \ldots, y''_{w_2-r_2-1}, z\}$ on z. If on the contrary, z occurs in (18), then we may obtain (17) by weakening over one of these two clauses. Which clause depends on whether z occurs positively or negatively. This completes the proof of the Theorem. $\square$

An immediate corollary of the proof of Theorem 5 and Theorem 8 is that every Resolution refutation of a a random 3CNF formula requires width $n/2(\ln n)^2$ with probability approaching 1 as n approaches ∞. Now, Ben-Sasson and Wigderson [BSW01] proved that if F is a 3CNF on n variables that has a Resolution refutation of length S, then F has a Resolution refutation of width $O\left(\sqrt{n \log S}\right)$. Thus, we obtain a slightly weaker bound than that of Chvátal and Szeméredi.

Corollary 2 *Let F be drawn from $\mathcal{F}_3^{n,6}$. Then, almost surely, F is unsatisfiable and every Resolution refutation of F requires size $2^{\Omega(n/(\ln n)^4)}$.*

From Theorem 7 we can also obtain Haken's lower bound. Indeed, there exist $(\Omega(n), 1/4)$-bipartite expanders of left-degree 3 on sets of size $n+1$ and n. Therefore, for such a bipartite expander G, every Resolution refutation of the set of clauses G-PHP requires width $\Omega(n)$. Since the number of variables of G-PHP is $O(n)$, we conclude that every Resolution refutation of G-PHP requires size $2^{\Omega(n)}$. From this, it is easy to obtain the same lower bound for PHP_n^{n+1} (see [BSW01] for details).

References

[ABE01] A. Atserias, M. L. Bonet, and J. L. Esteban. Lower bounds for the weak pigeonhole principle and random formulas beyond resolution. Accepted for publication in *Information and Computation*. A preliminary version appeared in ICALP'01, *Lecture Notes in Computer Science* 2076, Springer, pages 1005–1016., 2001.

[Ach01] D. Achlioptas. A survey of lower bounds for random 3-SAT via differential equations. *Theoretical Computer Science*, 265(1–2):159–185, 2001.

[AS00] D. Achlioptas and G.B. Sorkin. Optimal myopic algorithms for random 3-SAT. In *41st Annual IEEE Symposium on Foundations of Computer Science*, pages 590–600, 2000.

[BFU93] A. Z. Broder, A. M. Frieze, and E. Upfal. On the satisfiability and maximum satisfiability or random 3-CNF formulas. In *4th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 322–330, 1993.

[BH79] A. Blass and F. Harary. Properties of almost all graphs and complexes. *Journal of Graph Theory*, 3:225–240, 1979.

[BKPS99] P. Beame, R. Karp, T. Pitassi, and M. Saks. The efficiency of resolution and Davis-Putnam procedures. Submitted; final version of FOCS'96 and STOC'98, 1999.

[BSG01] E. Ben-Sasson and N. Galesi. Space complexity of random formulas in resolution. In *16th IEEE Conference in Computational Complexity*, 2001.

[BSI99] E. Ben-Sasson and R. Impagliazzo. Random CNF's are hard for the polynomial calculus. In *40th Annual IEEE Symposium on Foundations of Computer Science*, 1999.

[BSW01] E. Ben-Sasson and A. Wigderson. Short proofs are narrow–resolution made simple. *Journal of the ACM*, 48(2):149–169, 2001.

[CR79] S. Cook and R. Reckhow. The relative efficiency of propositional proof systems. *Journal of Symbolic Logic*, 44:36–50, 1979.

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

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

[Fag76] R. Fagin. Probabilities on finite models. *Journal of Symbolic Logic*, 41:50–58, 1976.

[Hak85] A. Haken. The intractability of resolution. *Theoretical Computer Science*, 39:297–308, 1985.

[JSV00] S. Janson, Y. C. Stamatiou, and M. Vamvakari. Bounding the undatisfiability threshold of random 3-SAT. *Random Structures and Algorithms*, 17(2):103–116, 2000.

[KV90] Ph. G. Kolaitis and M. Y. Vardi. 0-1 laws and decision problems for fragments of second-order logic. *Information and Computation*, 87:302–338, 1990.

[KV95] Ph. G. Kolaitis and M. Y. Vardi. On the expressive power of Datalog: tools and a case study. *Journal of Computer and System Sciences*, 51:110–134, 1995.

[KV00] Ph. G. Kolaitis and M. Y. Vardi. Conjunctive-query containment and constraint satisfaction. *Journal of Computer and System Sciences*, 61(2):302–332, 2000.

[Lyn80] J. F. Lynch. Almost sure theories. *Annals of Mathematical Logic*, 18:91–135, 1980.

[Pap85] C. H. Papadimitriou. A note on the expressive power of Prolog. *Bullentin of the EATCS*, 26:21–23, 1985.

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

[SS88] S. Shelah and J. Spencer. Zero-one laws for sparse random graphs. *Journal of the American Mathematical Society*, 1(1):97–115, 1988.

[Ull89] J. D. Ullman. Database and knowledge-base systems. *Computer Science Press,* 1989.