DETERMINISTIC SIMULATION OF PROBABILISTIC CONSTANT DEPTH CIRCUITS

Miklos Ajtai and Avi Wigderson

ABSTRACT

We explicitly construct, for every integer n and $\varepsilon \geqslant 0$, a family of functions (pseudorandom bit generators) $f_{n,\varepsilon}: \{0,1\}^{n^\varepsilon} \to \{0,1\}^n$ with the following property: for a random seed, the pseudorandom output "looks random" to any polynomial size, constant depth, unbounded fan-in circuit. Moreover, the functions $f_{n,\varepsilon}$ themselves can be computed by uniform polynomial size, constant depth circuits.

Some (interrelated) consequences of this result are given:

1. Deterministic simulation of probabilistic algorithms. The constant depth analogues of the probabilistic complexity classes RP and BPP are contained in the deterministic

Advances in Computing Research, Volume 5, pages 199–222.

ISBN: 0-89232-896-7

complexity classes $DSPACE(n^\varepsilon)$ and $DTIME(2^{n^\varepsilon})$ for any $\varepsilon \geqslant 0$.

2. Making probabilistic constructions deterministic. Some probabilistic construction of structures that elude explicit constructions can be simulated in the above complexity classes.
3. Approximate counting. The number of satisfying assignments to a (CNF or DNF) formula, if not too small, can be arbitrarily approximated in $DSPACE(n^\varepsilon)$ and $DTIME(2^{n^\varepsilon})$, for any $\varepsilon > 0$.

We also present two results for the special case of depth 2 circuits. They deal, respectively, with finding a satisfying assignment and approximately counting the number of satisfying assignments. For example, 3-CNF formulas with a fixed fraction of satisfying assignments, both tasks can be performed in polynomial time!

1. INTRODUCTION

The relationship between randomized and deterministic computation is a fundamental issue in the theory of computation. The results on this subject fall into the following categories.

1.1. Simulating Randomness by Nonuniformity

Adleman [Ad] showed that any language in RP can be computed by a polynomial size family of circuits. However, the proof is existential, and there is no known way of explicitly constructing these circuits. A similar result, for simulating probabilistic, polynomial size, constant depth circuits by nonuniform deterministic ones is due to Ajtai and Ben-Or [AB].

1.2. Simulating Randomness under an Unproven Assumption

Yao [Ya] has shown that if one way functions exist, then RP is contained in $DTIME(2^{n^\varepsilon})$, for any fixed positive ε. Note that the assumption is extremely strong, as it implies in particular that $P \neq NP \cap coNP$. Similar results are given in [FLS], who study the space complexity of the simulation, and [RT], who consider RNC instead of RP.

1.3. Simulating Randomness by Alternation

Sipser and Gacs [Si] showed that BPP is contained in Δ_2^p. Of course, the time or space complexities of languages in this class are unknown. A related result, due to Stockmeyer [St], is that approximate counting is in Δ_3^p.

1.4. Simulating Specific Randomized Algorithms

By a careful analysis of how randomness is used in a specific algorithm, one may be able to replace it by a deterministic construction. Such examples are the parallel algorithms in [Lu, KUW, KW]. Also related are explicit constructions of graphs with special properties, which can be found in [Ma] and [GG].

There were no explicit upper bounds on the deterministic simulation of any nontrivial class of probabilistic algorithms. In fact, there is no such simulation that does less than brute force enumeration of all possibilities for the random inputs.

We prove in this paper that probabilistic, polynomial size, constant depth, unbounded fan-in circuits can be simulated in $DSPACE(n^\varepsilon)$ [and hence also in $DTIME(2^{n^\varepsilon})$], for every fixed positive ε. This is done by generating a small set of pseudorandom binary strings, such that a randomly chosen one of them "looks random" to any polynomial size, constant depth circuit.

It is interesting to note that our "pseudorandom bit generator" is purely combinatorial, in contrast to the number theoretic generators used in cryptography (e.g., [Sh, BM, BBS]).

The proof that our generator "works" requires an intimate understanding of the structure of constant depth circuits. Such an understanding is drawn from the lower bound proof techniques for such circuits [Aj, FSS]. Moreover, these lower bounds are all "probabilistic" (or "nonconstructive"), and an essential part of building the generator is making them explicit. To this end we use the idea of "k-wise independent" random variables (e.g., see [ACGS, Lu, An, KUW]).

In Section 2 we give definitions and state our main theorem. In Section 3 we discuss applications of the main theorem, and in Section 4 we give the proof. In Section 5 we obtain refined results on depth 2 circuits, and discuss their applications.

2. DEFINITIONS AND THE MAIN THEOREM

A *circuit* C is a directed acyclic graph with node labels. The nodes of indegree zero are labeled with input variables, the nodes of outdegree zero are labeled with output variables, and the rest of the nodes are labeled from $\{AND, OR, NOT\}$. We put no bound on fan-in or fan-out.

The *size* of a circuit C, $s(C)$, is the number of nodes in it. The *depth of* C, $d(C)$, is the length of the longest input–output path. We say that C is an *(s,d)-circuit* if $s(C) \leqslant s$ and $d(C) \leqslant d$.

We shall be interested in families of circuits. Let s, d: $N \to N$ be functions. We say that $\{C_n\}$, $n = 1, 2, \ldots$ is an *(s,d)-family* if for all n, $s(C_n) \leqslant s(n)$, $d(C_n) \leqslant d(n)$. If $s = n^{O(1)}$, $d = O(1)$ then $\{C_n\}$ is a PC family (polynomial size, constant depth).

A family is uniform if there exists a Turing machine that on input n in unary, outputs a description of C_n, using only $O(\log n)$ space [Ru]. We shall mainly deal with one output circuit. Every such circuit C with n inputs computes a function C: $\{0,1\}^n \to \{0,1\}$ in a natural way. Define $p(C) = Pr[C(x) = 1]$, where $x \in \{0,1\}^n$ with uniform probability.

For inputs that are generated pseudorandomly we use the following. Let f: $\{0,1\}^m \to \{0,1\}^n$ be a function. Define $p_f(C) = Pr[C(x) = 1]$, where $x = f(y)$ and $y \in \{0,1\}^m$ with uniform probability.

Two important parameters measure the "goodness" of f as a pseudorandom bit generator for a circuit C. The natural one is $|p(C) - p_f(C)|$. Another parameter,, for which we get better bounds, is how small can $p(C)$ get so that still $p_f(C)$ does not vanish.

We can now state the main theorem. The present form of the Main Theorem is stronger than the one given in [AW], since the order of quantification has been changed.

Main Theorem. Let ε be fixed. Then there exists a family of functions $\{f_n$: $\{0,1\}^{n^\varepsilon} \to \{0,1\}^n\}$, $n = 1, 2, \ldots$, with the following properties:

- (i) $\{f_n\}$ can be computed by a uniform, PC family of circuits. (So in particular, $\{f_n\}$ can be computed in LOGSPACE.)
- (ii) Let l, d, u be fixed integers and $\{C_n\}$ be any (n^l, d) family of circuits. Then for every sufficiently large n,

(a) $|p(C_n) - p_{f_n}(C_n)| \leqslant n^{-\kappa}$
(b) for a fixed $\tau = \tau(l, d, \varepsilon)$, if $p(C_n) \geqslant 2^{-n^\tau}$, then $p_{f_n}(C_n) > 0$.

3. APPLICATIONS

The applications are given not necessarily in order of importance, but rather in order of notational convenience. All the applications are based on the fact that we can get a fairly good approximation of the output behaviour of C_n by "testing" it on only 2^{n^ε} inputs.

The following notation will be used often. Let g: $N \to [0, 1]$ be a function. We say that g is polynomially small if $g(n)^{-1} = n^{O(1)}$. We say that g is subexponentially small if $g(n)^{-1} = o(2^{n^\varepsilon})$, for every fixed $\varepsilon > 0$.

3.1. Approximate Counting

Let the number of satisfying assignments to a (*CNF* or *DNF*) formula F be $\# F$. Computing $\# F$ from F is $\# P$ complete. It is not known whether $\# P$ is in the polynomial hierarchy. An easier problem is approximate counting, which is in this case to find an integer in the range $[(1 + \beta)^{-1} \# F, (1 + \beta) \# F]$. Call this β approximation. For any polynomially small β, β approximation was shown to be in Δ_3^p by Stockmeyer [S]. No explicit deterministic upper bounds were known for approximate counting. Let $p(F) = \# F/2^n$ be the fraction of satisfying assignments of F, where n is the number of variables in F.

COROLLARY 1. Consider formulas F with polynomially small $p(F)$. Then for every fixed ε and every polynomially small β, the β approximation problem for F is in $DSPACE(n^\varepsilon)$ [and hence also in $DTIME(2^{n^\varepsilon})$].

Proof (*sketch*). F is a polynomial size depth 2 circuit. In $DSPACE(n^\varepsilon)$ all the "seeds" y of f_n can be generated, $f_n(y)$ computed and tested on F. The output is $(\sum_{|y| = n^\varepsilon} F[f_n(y)])2^{n - n^\varepsilon}$. By (*ii*), part (*a*) of the main theorem, it is the desired approximation.

3.2. Easy Cases of Satisfiability

If we are just interested in finding a satisfying assignment to a formula F, the result above can be improved. In [VV], Valiant and

Vazirani showed that finding a satisfying assignment in formulas with exactly one such assignment is essentially as hard as the general case. The following result, which complements theirs, says that if the number of satisfying assignments is large enough, then satisfiability becomes easier.

COROLLARY 2. Consider formulas F with subexponentially small $p(F)$. Then for any $\varepsilon > 0$ a satisfying assignment of F can be found in $DSPACE(n^{\varepsilon})$ [and $DTIME(2^{n^{\varepsilon}})$].

This result follows from (*ii*), part (*b*) of the main theorem.

3.3. Making Probabilistic Constructions Deterministic

Following Sipser [Si], we define a probabilistic construction to be a language $L \subseteq \{0,1\}^* \times \{0,1\}^*$ with the property that if $(u, v) \in L$, then $Pr[(u, x) \in L] \geqslant 1/2$, where x is uniformly chosen with $|x = |v|$. (u usually gives the size of the required object in unary, and then a random object of the right size has the desired properties.) The deterministic construction problem for L is, on input u, to generate v s.t. $(u, v) \in L$.

If $L \in \sum_i^p$, Sipser calls it a $\sum_i^p$ construction. He shows that if L is a $\sum_i^p$ construction, then the deterministic construction problem for L is, is in $\sum_{i+2}^p$. (An analogous statement is true for $\prod_i^p$.)

Note that $L \in \sum_i^p$ or $L \in \prod_i^p$ means that L can be recognized by a uniform family of constant depth (but possibly exponential size) circuits. We say the L is a PC construction if it can be recognized by a uniform PC family of circuits.

COROLLARY 3. If L is a PC construction, then for any $\varepsilon > 0$ the deterministic construction problem for L is in $DSPACE(n^{\varepsilon})$ [and in $DTIME(2^{n^{\varepsilon}})$].

Note that the uniformity is needed for our deterministic machine to generate the circuit recognizing constructions of size $|u|$, where u is the input.

3.4. Deterministic Simulation of Probabilistic Constant Depth Circuits

A probabilistic circuit C is one with "real" input variables, z, and random input variables, x. If $|z| = n$ and $|x| = m[=m(n)]$,

C computes a function $C\colon \{0,1\}^{n+m} \to \{0,1\}$. Let $p[C(z)] = Pr[C(z,x) = 1]$ when $x \in \{0,1\}^m$ with uniform probability. The idea of recognizing languages by probabilistic circuits is that the behaviour of $p[C(z)]$ will depend on whether z is in the language or not.

We define two families of complexity classes, $PC1(\alpha)$, $PC2(\alpha)$ where PC refers to polynomial size, constant depth, 1 and 2 refers to whether we allow one or two sided errors, and α is the "accuracy" (in general $\alpha\colon N \to [0,1]$ is a function). A language $L \subseteq \{0,1\}^*$ is in $PC1(\alpha)$ if there exists a uniform PC family $\{C_n\}$ s.t. for every n and every $z \in \{0,1\}^n$ we have

$$z \in L \to p[C(z)] \geqslant \alpha(n) \text{ and } z \notin L \to p[C(z)] = 0.$$

A language $L \subseteq \{0,1\}^*$ is in $PC2(\alpha)$ if there exists a uniform PC family $\{C_n\}$ s.t. for every n and every $z \in \{0,1\}^n$

$$z \in L \to p[C(z)] \geqslant 1/2 + \alpha(n)$$

$$z \notin L \to p[C(z)] \leqslant 1/2 - \alpha(n).$$

COROLLARY 4. For every fixed $\varepsilon > 0$ we have

(*i*) for every subexponentially small function α, $PC1(\alpha) \subseteq DSPACE(n^\varepsilon)$

(*ii*) for every polynomially small function α, $PC2(\alpha) \subseteq DSPACE(n^\varepsilon)$.

4. PROOF OF THE MAIN THEOREM

The key notion is that of approximating a circuit. A given circuit will undergo a series of simplifications, each restricting the inputs, that will change the output behaviour by only tiny amounts.

The proof has two logical parts. In Part I we show how to approximate $n^{1-\varepsilon}$ input bits of a PC circuit by only $O(\log n)$ bits. In Part II we show how to iterate this construction, adding only a constant to the depth and a polynomial to the size.

4.1. Part I: Sketch of the Proof

As we mentioned, the task in this part is to replace $n^{1-\varepsilon}$ random bits by $O(\log n)$ random bits without affecting the output behavior of the circuit by much, and implement this change by a PC circuit.

Suppose that C is a constant depth polynomial size circuit with the set of variables $A = \{x_1, \ldots, x_n\}$. First we show that if W is a random subset of A with $n^{1-\varepsilon}$ elements and we substitute random values for the variables in $A - W$ then with high probability the resulting circuit will actually depend only on t variables where t is a constant. [See Definition 1. "W is (t, τ)-local."]

If we know such a set W then we can define a pseudorandom input f for C in the following way. Let $f|_{A-W}$ and $f|_W$ be independent, $f|_{A_W}$ uniform on the set of all possible assignments on $A - W$ and let $f|_W$ be uniform on all subsets of W with t elements. [See Definitions 2 and 3; "Y is $(|W|, t)$-uniform."] We will show how to generate such an $f|_W$ only from $O(\log n)$ random bits. For such an f we have that $|Pr[C(x) = 1] - Pr[C(f) = 1]| < n^{-u}$ where u is a large constant.

Unfortunately f still depends on the subset W. To pick a random W with $n^{1-\varepsilon}$ elements requires too many random bits. We will show however that a W with the required properties can be generated from only $O(\log n)$ random bits. To make this part of the argument clearer, first we show those combinatorial properties of a random subset W that guarantee that the circuit we get from C by substituting random bits for the variables in $A - W$ depends only on t elements.

If we consider only depth 2 circuits the essential property of W is the following: if a small subset V of A is given $[|V| \leqslant n^{\beta}, \beta + (1-\varepsilon) < 1]$ then with high probability W will intersect it in a set of constant size. (See Definition 4. "Small intersection property.") This property will imply that if a polynomial family of V's are given then still with high probability W will intersect each of them in a set of constant size. This property will make it possible to replace large conjunctions (or disjunctions) by small ones.

For depth d circuits we have to iterate this argument, so W must be included in a sequence of sets $X_0 \supseteq X_1 \supseteq \cdots \supseteq X_d = W$ so that each X_i has the small intersection property in X_{i-1}. (See Definition 5. "d-iterated small intersection property.)

Now we can define the pseudorandom input f from the $n - n^{1-\varepsilon}$ random bits, which specifies its value on $A - W$, and from the $O(\log n)$ random bits, which describes a W with the d-iterated small intersection property and another $O(\log n)$ random bits, which give a $(|W|, t)$ uniform $f|_W$. So altogether from $n - n^{1-\varepsilon} + O(\log n)$ random bits we defined a pseudorandom input f so that $|p(C) - p_f(C)| < n^{-u}$. In Part II we will iterate this argument to decrease the number of necessary random bits to n^{ε}.

We need some notation. Let C be a circuit on n variables $A = \{x_1, \ldots, x_n\}$. For any subset $Y \subseteq A$ and a binary vector v of length $|Y|$, let $C_{Y,v}$ denote the circuit obtained from C by assigning the values v to the variables in Y, in the natural order. If v is chosen uniformly at random then denote by C_Y the random variable $C_{Y,v}$. Let Z be a random variable taking values from $\{0,1\}^A$, then denote $Pr[C(Z) = 1]$ by $p[C(Z)]$.

Proposition 1 asserts that for a random set of all but $n^{1-\epsilon}$ of the input variables and a random assignment to them, the resulting circuit will depend only on a constant number of inputs (although it has $n^{1-\epsilon}$ of them).

Definition 1. We say that a circuit C depends on t variables if there exists a subset $T \subseteq A$ of size t s.t. for every assignment to the variables in T, the resulting circuit is constant. We denote the minimum such t by $t(C)$ and some T of this cardinality by $T(C)$.

If t is an integer and $\tau > 0$ the set $W \subseteq A$ is called (t, τ)-local if

$$Pr[t(C_{A-W}) \leqslant t] \geqslant 1 - 2^{-n^{\tau}}.$$

Proposition 1. Let d, l, u be positive integers and $\varepsilon > 0$. Then there exists an integer t so that if n is sufficiently large, C is a (n^l, d) circuit with n input variables A, and $W \subseteq A$ is random with $|W| = n^{1-\epsilon}$, then $Pr[t(C_{A-W}) > t] \leqslant n^{-u}$ where we take the probability over the product space "choose W, then choose assignment on $A - W$."

Moreover, there exists a $\tau > 0$ depending only on d, l, u, ε so that with probability at least $1 - n^{-u}$ the set W is (t, τ)-local. (Here the probability is over choices of W only.)

The proof of Proposition 1 is an inductive argument on the depth of the circuit, similar in flavor to the lower bound proofs in [Aj], [FSS], [Ha], and [Ya]. In fact, the first part of Proposition 1 appears with a different proof in [Aj]. The inductive step is based on a property of depth 2 circuits, which will be given in Theorem 1 (Section 5).

We do not prove Proposition 1 now since we later (Proposition 2) will prove a stronger version of it, namely we can pick W randomly from a uniformly given polynomial size family of subsets of A. This remark also relates to Corollary 5.

Note that different choices of v may result in different subsets $T(C_{A-W,v})$ of W that the resulting circuit depends on. However, Proposition 1 tells us that the output distribution of C will essentially remain unchanged if instead of assigning random values to the variables in W (not in $W-A$), we use assignments that "look random" only on t subsets of W. This motivates the next definitions and corollary.

DEFINITION 2. A random variable $Z = Z_1, \ldots, Z_m$ with $Z_i \in \{0,1\}$ is said to be (m,t,p)-uniform if for all $1 \leqslant i \leqslant m$ $Pr(Z_i = 1) = p$ and for every t-subset I of $\{1,\ldots,m\}$ the variables $\{Z_i \mid i \in I\}$ are mutually independent. When $p = 0.5$ we say that Z is (m,t)-uniform.

The key fact about (m,t)-uniform sequences is that they can be simply generated from only $t \log m$ random bits by PC circuits, using polynomials over finite fields. The explicit construction will be given later.

DEFINITION 3. Let $W \subseteq A$ and integer t be fixed. A random variable $Y \in \{0,1\}^n$ is called (W,t)-uniform if $Y|_W$ is $(|W|,t)$-uniform, $Y|_{A-W}$ is uniform, and these two restricted random variables are independent.

Corollary 5 below follows easily from the above definitions and Proposition 1.

COROLLARY 5. Let $d, l, u, \varepsilon, t, \tau$ be as in Proposition 1. For each possible W of size $n^{1-\varepsilon}$ let $Y^{(W)} \in \{0,1\}^n$ be an arbitrary (W,t)-uniform random variable. Then

(1) If W is chosen uniformly at random, then we have $|p(C) - p[C(Y^{(W)})]| \leqslant n^{-\nu}$.
(2) If W is (t,τ)-local, then $|p(C) - p[C(Y^{(W)})]| \leqslant 2^{-n^{\tau}}$.
(3) If W is (t,τ)-local then there is an evaluation w on W so that $Pr(Y^{(W)}|_W = w) > 0$ and $Pr(C_{W,w} = 1) \geqslant p[C(Y^{(W)})] - 2^{-n^{\tau}}$.

Proof. Fix v in $\{0,1\}^{A-W}$, let $C' = C_{A-W,v}$ (on inputs from W), and consider $p(C')$ vs. $p(C', Y^{(W)}|_W)$. In the first we consider a uniform distribution on inputs to C' and in the second only a t-uniform distribution. The circuit reacts identically if $t(C') \leqslant t$ hence we get errors only if $t(C') > t$. By Proposition 1, for a random W and v, this happens with probability smaller than n^{-u}. (1)

For a (t,τ)-local set W, this happens with probability smaller than $2^{-n^{\tau}}$. (2)

(3) follows from (2) by averaging over the possible assignments w to W in the random variable $Y^{(W)}$ [replace $p[C(Y^{(W)})]$ by $p(C)$].

We are now in the situation where, if given a (t,τ)-local set W, we can replace its $n^{1-\varepsilon}$ random input bits by $O(\log n)$ random bits. Corollary 5 shows that most W will work, but to specify a random set W we need as many as $n^{1-\varepsilon}\log n$ bits. Our next step will be to generate (t,τ)-local sets W pseudorandomly, using only $O(\log n)$ random bits. This is done by extracting from the proof of Proposition 1 only the essential properties of the random variable W that are actually used.

Definition 4. Let X be a random variable whose values are subsets of a set A of size n. We say that X has the small intersection property with parameters α, β, t if for every set $V \subseteq A$ with $|V| \leqslant n^{\beta}$ we have that if $s \leqslant t$, then

$$Pr(|V \cap X| \leqslant s) \geqslant 1 - n^{-(1-\alpha-\beta)s}.$$

Definition 5. The random variable $X \subseteq A$ has the d-iterated small intersection property with parameters α, β, t if there exists a sequence of random variables $X_1, \ldots, X_d$ so that $X = X_d$, $X_{i+1} \subseteq X_i$ and for any possible fixed values $B_1, \ldots, B_i$ of the variables $X_1, \ldots, X_i$ we have that X_{i+1} with the condition $X_1 = B_1, \ldots, X_i = B_i$ has the small intersection property with parameters α, β, t.

Proposition 2. The consequences of Proposition 1 and Corollary 5 hold if we replace a randomly chosen W of size $n^{1-\varepsilon}$ by a random variable W that has the d-iterated small intersection property with parameters $1-\varepsilon$, $\varepsilon/2$, t

The proof of Proposition 2 is based on the following Lemmas, 1 and 2.

Before stating the lemmas, we state Theorem 1, which is proved in Section 5. Intuitively, it shows that a depth 2 circuit of small bottom fan-in is almost completely determined by a small subset of its input variables.

THEOREM 1. Let C be a depth 2 circuit of bottom fan-in $\leqslant k$ with input variables in A. Then for every $r \geqslant 2^{2k^3}$ there exists a subset $Q \subseteq A$ s.t.

(1) $|Q| \leqslant r^{2k^2}$
(2) $Pr(C_Q \not\equiv 0, 1) \leqslant 2^{-r}$.

Furthermore, if $|A| = n$, the set Q can be found in time $O(n^{k^2})$.

LEMMA 1. Let k, ε be fixed. For any z if n is sufficiently large, C is a depth 2 circuit with bottom fanin at most k; W is a random variable with the $(1 - \varepsilon, \varepsilon/2, z)$ small intersection property and $t \leqslant z$ then with probability at least $1 - n^{-\varepsilon t/2}$ the set W satisfies the following inequality:

$$Pr[t(C_{A-W}) \leqslant t] \geqslant 1 - 2^{-n^{\varepsilon/5k^2}}.$$

Proof of Lemma 1. Apply Theorem 1 with $r = n^{\varepsilon/4k^2}$ to obtain the set Q. By (1) of the theorem, $|Q| \leqslant r^{2k^2} = n^{\varepsilon/2}$, and since W has the small intersection property, $Pr(|Q \cap W| \leqslant t) \geqslant 1 - n^{-\varepsilon t/2}$.

Now fix W so that $|W \cap Q| \leqslant t$. By (2) of the theorem at most $2^{|Q|-r}$ of the assignments to Q do not determine the value of C. Hence, at most $2^{|Q|-r}$ of the assignments to $Q - W$ will have an extension in $Q \cap W$ that does not fix C, and since these are chosen randomly in C_{A-W} and $|Q - W| \geqslant |Q| - t$ we have $Pr[t(C_{(A-W)}) > t] \leqslant 2^{|Q|-r}/(2^{|Q|-t}) = 2^{-r+t} = 2^{-(n^{\varepsilon/4k^2}+t)}$. If n is sufficiently large then $n^{\varepsilon/4k^2} - t \geqslant n^{\varepsilon/4k^2} - z \geqslant n^{\varepsilon/5k^2}$ so the probability is not greater than $2^{-n^{\varepsilon/5k^2}}$.

Using Lemma 1, we can reduce the depth of a PC circuit C (as in [Aj], [FSS]) by applying it to all the bottom depth 2 circuits of C. Once each of these depends only on t inputs, we change it from CNF to DNF or vice vera without blowing the size up by more than 2^t (a constant) and hence reduce the number of alternations (depth) of C by 1. This is the essence of Lemma 2, and since a pseudo-random W is good enough for Lemma 1, it suffices also for Lemma 2.

LEMMA 2. For all d, l, k, $\varepsilon > 0$ there is a t and a $\tau > 0$ so that for any $z \geqslant t$ if n is sufficiently large and C is an (n^l, d) circuit with bottom fanin at most k, and W has the small intersection property with parameters $1-\varepsilon$, $\varepsilon/2$, z then with probability at least $1-n^{-[(\varepsilon t/2)-t]}$ the set W will satisfy $Pr(C_{A-W}$ is a depth $d-1$ circuit with bottom fan-in $t) \geqslant 1-2^{-n^{\varepsilon}}$.

Now to prove Proposition 2, we simply apply Lemma 2 d times to the circuit (again, as in [Aj], [FSS], [Ya], [Ha]). The resulting circuit (with high probability) depends only on a constant number of inputs, which implies the proposition.

Note that in Definition 4 only intersections of cardinality t or less are important. From this it is easy to deduce:

LEMMA 3. If W is an $(n, n^{-\varepsilon}, t)$-uniform random variable, then $X = \{i \mid W(i) = 1\}$ has the small intersection property with parameters $1-\varepsilon, \varepsilon/2, t$.

Proof of Lemma 3. Let $V \subseteq A$ with $|V| \leqslant n^{\varepsilon/2}$, $s \leqslant t$. If $|V \cap X| \geqslant s$ then there are distinct $v_1, \ldots, v_s \in V \cap X$. The number of s-tuples $v_1, \ldots, v_s$ is $\binom{|V|}{s} \leqslant \binom{n^{\varepsilon/2}}{s}$, and for any fixed s-tuple $v_1, \ldots, v_s$ we have $Pr(v_1, \ldots, v_s \in X) \leqslant (n^{-\varepsilon})^s$ since $s \leqslant t$ and W is $(n, n^{-\varepsilon}, t)$-uniform. So $Pr(|V \cap X| \geqslant s) \leqslant \binom{n^{\varepsilon/2}}{s}(n^{-\varepsilon})^s \leqslant n^{s(\varepsilon/2)} n^{-s\varepsilon} = n^{-[1-(1-\varepsilon)-\varepsilon/2]s}$.

Again such a random variable can be constructed from $t \log n$ random bits by PC circuits. To get a random variable with the d-iterated small intersection property one can use d independent constructions as above, which require only $dt \log n$ random bits. We will give the construction in detail in Part II.

To summarize the first part of the proof, we have shown how to replace $n^{1-\varepsilon}$ random bits by $O(\log n)$ random bits, thus reducing the number of inputs by roughly $n^{1-\varepsilon}$ without substantially affecting the output probability of the circuit. This was done by first using $O(\log n)$ bits to specify a pseudorandom set W of size $n^{1-\varepsilon}$ of inputs. Then use other $O(\log n)$ bits to create a pseudorandom assignment to variables in W. The remaining $n - n^{1-\varepsilon}$ inputs receive truly random assignments.

4.2. Part II

At this point it is natural to iterate the construction roughly n^{ε} times. However, this presents some difficulties. For example, if we implement the construction in the first part, the depth of the circuit increases by a constant, and so we cannot repeat that more than a constant number of times. Another problem that bounds the number of iterations is that we must keep the circuit polynomial in the number of remaining inputs, so we have to stop when at least a polynomial fraction remains.

Conceptually performing this iterative process, we obtain a sequence of roughly n^{ε} pseudorandom subsets of variables, that together with the remaining part (of size roughly n^{ε}) form a partition of the set of variables. To each pseudorandom subset we assign (independently) a pseudorandom assignment [requiring a total of $O(n^{\varepsilon}\log n)$ bits], and to the remaining subset assign random variables (only n^{ε}).

In order to perform this process in constant depth, we shall generate all parts in the partition together with their assignments simultaneously. We first define the partition assignment pair abstractly, as random variables, and then show how they can be generated from n^{ε} random bits.

DEFINITION 6. Let d, t be integers and $\delta > 0$. For every n and $0 \leqslant \mu \leqslant n$ define $\langle F, P\rangle$ to be a (d, t, δ)-fooling pair of random variables if the following conditions hold:

(1) Each value of F is a $0,1$ assignment to the variables in A, $|A| = n$.

(2) Each value of $P = \langle P_0, \ldots, P_\mu\rangle$ is a sequence of subsets of A so that $P_0, \ldots, P_\mu$ form a partition of A.

(3) For all $0 \leqslant i < \mu$ if $A_0, \ldots, A_{i-1}$ are fixed disjoint subsets of A then the random variable P_i with the condition $P_0 = A_0, \ldots, P_{i-1} = A_{i-1}$ has the d-iterated small intersection property with parameters $1 - 2\delta, \delta, t$.

(4) For all $0 \leqslant i \leqslant \mu$ if $A_0, \ldots, A_i$ are fixed subsets of A then with the condition $P_0 = A_0, \ldots, P_i = A_i$ the random variables $F|_{A_0}, \ldots, F|_{A_i}$ are independent.

(5) For all $0 \leqslant i < \mu$ if $A_i \subseteq A$ then $F|_{A_i}$ with condition $P_i = A_i$ is an $(|A_i|, t)$-uniform random variable.

(6) There is a set $A_\mu \subseteq A$ so that $|A_\mu| = n^{\delta}$, $Pr(P_\mu = A_\mu) \geqslant 1 - 2^{-n^{\delta}}$, $Pr(A_\mu \subseteq P_\mu) = 1$, and $F|_{A_\mu}$ has a uniform distribution over all possible assignments on A_μ.

The technical properties of the fooling pair guarantee that it fools any PC circuit with the appropriate parameters.

PROPOSITION 3. For all d, l, u, δ there exists a t such that for all sufficiently large n, $\mu < n$ and a (d, t, δ)-fooling pair $\langle F, P \rangle$ we have the following.

(1) For every (n^l, d) circuit C with n inputs, $|p(C) - p[C(F)]| \leqslant n^{-u}$.

(2) There exists a $\tau > 0$ depending only on d, l, u, δ so that $p[C(F)] \geqslant p(C) - \mu 2^{n^{-\tau}}$.

The proof of this Lemma will be by induction, which will show that the simultaneous construction definition of the fooling partition assignment pair actually simulates the natural iterative construction. For this we need the following definition and Lemmas 4 and 5.

DEFINITION 7. For all $0 \leqslant i \leqslant \mu$ let Y_i be the random assignment to the variables in A that coincide with F on $\bigcup_{j<i} P_j$ and take random values uniformly and independently of $\langle F, P \rangle$ on $A - \bigcup_{j<i} P_j$.

LEMMA 4. For all but a fraction n^{-u-2} of the values B that P may take, and for all $0 \leqslant i \leqslant \mu - 1$ we have $|p[C(Y_i)] - p[C(Y_{i+1})]| \leqslant n^{-u-2}$, when these probabilities are conditioned by the event $P = B$.

Note that, conditioned on the event $P = B$ we have $Pr[C(Y_0) = 1] = p(C)$ and provided that $P_\mu = A_\mu$ we have $Pr[C(Y_\mu) = 1] = Pr[C(F) = 1]$ so, according to property (6) of a fooling pair, Lemma 4 implies part (1) of Proposition 3.

Proof of Lemma 4. In the following proof all probabilities are considered with the condition $P = B$. Let $F_i = F|_{\bigcup_{j<i} P_j}$. Then for every value B that P may take $Pr[C(Y_i) = 1] = \sum_f Pr(F_i = f) Pr[C(Y_i) = 1 | F_i = f]$ where f takes all of the possible values for F_i.

Suppose now that f is fixed. We may consider C as a circuit with the variables $A - \bigcup_{j<i} P_i$ by evaluating the remaining variables according to f. We will denote this circuit by D. According to the definition of a fooling pair, P_i, has the d-iterated small intersection property with parameters $1 - 2\delta, \delta, t$ (even if F_i is fixed). Proposition 2 implies that for all but a fraction n^{-u-4} of values B_i that P_i

may take we have that the set B_i is (t, τ)-local with respect to the circuit D. Therefore (2) of Corollary (5) implies that for all but a fraction $n^{-\nu-3}$ of values B_i that P_i may take, given the event $F_i = f$ we have

$$|Pr[C(Y_i) = 1] - Pr[C(Y_{i+1}) = 1]| \leqslant n^{-\nu-3}.$$

Since $\mu < n$ this implies that for all but a fraction $n^{-\nu-2}$ of the values B that P may take the inequality holds for all $i = 0, \dots, \mu - 1$.

LEMMA 5. For all but a fraction $n^{-\nu-2}$ of the values $B = \langle B_0, \dots, B_\mu \rangle$ that P can take and for all $0 \leqslant i < \mu - 1$ if f is an assignment with $Pr(F_i = f | P = B) > 0$, then there exists an extension f' of f to $\bigcup_{j \leqslant i} B_j$ so that

$$Pr(F_i = f' | P = B) > 0, \text{ and}$$

$$Pr[C(Y_{i+1}) = 1 | P = B, F_{i+1} = f']$$
$$\geqslant Pr[C(Y_i) = 1 | P = B, F_i = f] - 2^{-n^{\epsilon}}.$$

The proof of this Lemma is essentially the same as the proof of Lemma 4, only in the last step we use property (3) from the modified form of Corollary 5 as described in Proposition 2. Part (2) of Proposition 3 follows from Lemma 5.

Proposition 4 deals with the explicit construction of a fooling pair from a small number of random bits.

PROPOSITION 4. Let d, t be integers $\delta > 0$. Then for every large enough n a fooling pair $\langle F_n, P_n \rangle$ can be constructed with $\mu = [n^{(d+1)\delta}]$, by a LOGSPACE uniform PC family of circuits, given as inputs $2d(t+1)(\mu+1)\log_2 n + n^{\delta}$ random bits.

DEFINITION 8. (1) We will denote the finite field with q elements by K_q. We suppose (without the loss of generality) that n is a power of two, $n = 2^h$ and $K_n = K_2[\gamma_n]$ (where γ_n is given uniformly) and so the elements $1, \gamma_n, \dots, (\gamma_n)^{h-1}$ form a basis of the vectorspace K_n over K_2. For each $x \in K_n$ let $\bar{x}$ denote the sequence of coefficients of the representation of x in this basis. (There is no difficulty with giving the bases of K_n in LOGSPACE since an irreducible polynomial over K_2 of degree h can be found by the brute force method in LOGSPACE.)

(2) Let $g_n(x_1, \ldots, x_{u(n)}, y_1, \ldots, y_{v(n)})$ be a function with $u(n) + v(n)$ variables where each x_i can be an element of K_n and each y_j can be an integer between 0 and 2^{h-1}. We say that g_n can be uniformly computed by a family of (n^{c_1}, c_2) circuits if there are absolute constants c_1, c_2 and a uniform family of (n^{c_1}, c_2) circuits C_n with $[u(n) + v(n)]h$ inputs and h outputs so that for any sequence $x_1, \ldots, x_{u(n)}, y_1, \ldots, y_{v(n)}$ from the domain of g if the input of C_n is $\bar{x}_1, \ldots, \bar{x}_{u(n)}, \bar{y}_1, \ldots, \bar{y}_{v(n)}$ (where $\bar{y}_j$ is the binary representation of the number y_j) the output is $\bar{g}(x_1, \ldots, x_{u(n)}, y_1, \ldots, y_{v(n)})$.

LEMMA 6. The following functions can be computed by an (n^{c_1}, c_2) circuit:

(1) $g(x_1, x_2)$ or $g(x, y)$ for any $g \in LOGSPACE$.
(2) $x_1 x_2$.
(3) x^y.
(4) $x_0 + x_1 + \cdots + x_{h-1}$.
(5) $x_0 + x_1 x_h + x_2 x_h^2 + \cdots + x_{h-1}(x_h)^{h-1}$.

Proof. (1) If $\bar{x}_1 = \langle \alpha_0, \ldots, \alpha_{h-1} \rangle$, $\bar{x}_2 = \langle \beta_0, \ldots, \beta_{h-1} \rangle$, $\bar{g}(x_1, x_2) = \langle \gamma_0, \ldots, \gamma_{h-1} \rangle$ then $\gamma_i = \bigvee' \bigwedge_{j=0}^{h-1} (\alpha_j \leftrightarrow a_j \wedge \beta_j \leftrightarrow b_j \wedge \gamma_j \leftrightarrow d_j)$ where $\bigvee'$ is taken for all $a, b, d \in K_n$ with $d = g(a, b)$ and $\bar{a} = \langle a_0, \ldots, a_{h-1} \rangle$, $\bar{b} = \langle b_0, \ldots, b_{h-1} \rangle$, $\bar{d} = \langle d_0, \ldots, d_{h-1} \rangle$.

(2) and (3) follows immediately from (1).

(4) If $\alpha_{i,0}, \ldots, \alpha_{i,h-1}$ are the coefficients of x_i and $\beta_0, \ldots, \beta_{h-1}$ are the coefficients of $g(x_0, \ldots, x_{h-1})$ is the basis $1, \gamma, \ldots, \gamma^{h-1}$ then $\beta_j = \bigvee'[(\alpha_{0,j} \leftrightarrow a_0) \wedge \cdots \wedge (\alpha_{h-1,j} \leftrightarrow a_{h-1})]$ where the disjunction $\bigvee'$ is taken for all 0, 1 sequences $a_0, \ldots, a_{h-1}$ with $a_0 + \cdots + a_{h-1} \equiv 1 \pmod 2$. Since there are only $n/2$ such sequences the size of the circuit is polynomial in n.

(5) follows from (4), (3), and (2).

DEFINITION 9. If $x \in K_n$ let $int(x)$ denote the integer whose binary representation is the same as the sequence of coefficients of x in the basis $1, \gamma, \ldots, \gamma^{h-1}$ [that is $\bar{x} = \overline{int(x)}$]. Conversely if y is an integer between 0 and $n - 1$ then let $fld(y)$ be the element of K_n with $int[fld(y)] = y$.

(2) Suppose i, t are integers $0 \leqslant i, t < n$. Let us define a random variable $Z_{i,t}^n = \langle z_0, \ldots, z_{n-1} \rangle$ in the following way. Take a random polynomial of degree at most $t - 1$ $f(x) = a_0 + a_1 x + \cdots + a_{t-1} x^{t-1}$ over K_n, so that $a_0, \ldots, a_{t-1}$ are chosen uniformly

and independently from K_n. For each $0 \leqslant j < n$ let $z_j = 1$ iff $int[fld(j)] < i$.

(3) We define a random variable $S^n_{i,t,k}$ where i, t are as in the previous definition and k is a positive integer, whose values are of $\{0, \ldots, n-1\}$. Let $\langle s^1_0, \ldots, s^1_{n-1}\rangle, \ldots, \langle s^k_0, \ldots, s^k_{n-1}\rangle$ be k independent random values of the random variable $Z^n_{i,t}$. For each $0 \leqslant j < n$ let $j \in S^n_{i,t,k}$ iff $s^1_j = s^2_j = \cdots = s^k_j = 1$.

LEMMA 7. (1) If $0 \leqslant i, t < n$ then $Z^n_{i,t}$ is an $(n, t, i/n)$-uniform random variable.

(2) If $i = n^\alpha$ then $S^n_{i,t,k}$ satisfies the k-iterated small intersection property with parameters $(1 - \alpha, \alpha/2, t)$.

(3) $S^n_{i,t,k}$ has the following property: for each $0 \leqslant j < n$ we have $Pr(j \in S^n_{i,t,k}) = (i/n)^k$.

(4) If $i(n), t(n), k(n) \in LOGSPACE$ and $i(n), k(n) < n$, $0 \leqslant i < n$, $0 \leqslant t(n) \leqslant \log_2 n = h$ then there exists uniform families C_n, D_n of (n^{c_1}, c_2) circuits where c_1, c_2 are absolute constants, which realizes $Z^n_{i(n),t(n)}$ and $S^n_{i(n),t(n),k(n)}$. More precisely C_n has th inputs, D_n has kth inputs and both have n outputs, and if we take their input randomly (uniformly) then the output sequence of C_n has the same distribution as $Z^n_{i(n),t(n)}$ and the output sequence of D_n has the same distribution as the characteristic function of $S^n_{i(n),t(n),k(n)}$.

Proof of Lemma 7.

(1) See e.g. [KUW].

(2) Part (1) and Lemma 3 implies the assertion for $k = 1$. For an arbitrary k our statement follows from the trivial fact that the intersection of k independent random variables with the small intersection property with parameters $(1 - \alpha, a/2, t)$ has the k-iterated small intersection property with the same parameters.

(3) For $k = 1$ our statement is equivalent to the following: if f is a random polynomial of degree at most t in $K_n[x]$ then for any fixed $b \in K_n$ $Pr\{int[f(b)] < i\} = i/n$. This assertion follows from the fact that $f(b)$ has a uniform distribution in K. For an arbitrary k our assertion follows from the case $k = 1$ and the independence of the sequences $\langle s^j_0, \ldots, s^j_{n-1}\rangle$ in the definition of $S^n_{i,t,k}$.

(4) Follows from (5) of Lemma 6, since it guarantees that the random polynomial in the definition of $Z^n_{i,t}$ can be evaluated by a (n^{c_1}, c_2) circuit.

DEFINITION 10. If C is a circuit with n outputs then let $seq(C) = \langle s_0, \ldots, s_{n-1} \rangle$ be a random variable whose values are the output sequences of C if the input of C is taken randomly and uniformly from the set of all possible inputs and let $set(C) = \{0 \leq j < n \mid s_j = 1\}$.

Proof of Proposition 4. Assume $A = \{0, 1, \ldots, n-1\}$. Let $\delta > 0$, d, t be fixed. Suppose that n is sufficiently large and $\mu = [n^{(d+1)\delta}]$. Let $D_0, D_1, \ldots, D_\mu$ be disjoint (n^{c_1}, c_2) circuits with $d(t+1)\log_2 n$ inputs and n outputs, so that for all $i = 0, \ldots, \mu$ the random variable $set(D_i)$ satisfies the d-iterated $(1-\delta, \delta/2, t)$ small intersection property and for any $r \in A$ $Pr[r \in set(D_i)] \geq (n^{1-\delta}/n)^\delta$ (where c_1 and c_2 are absolute constants). Note that Lemma 7 implies the existence of such circuits. Let $G_0, G_1, \ldots, G_{\mu-1}$ be disjoint (n^{c_3}, c_4) circuits with $(t+1)\log_2 n$ inputs and n outputs (where c_3, c_4 are absolute constants) so that for each $0 \leq i < \mu$ $seq(G_i)$ is (n, t)-uniform. Lemma 7 guarantees the existence of such circuits. Let G_μ be a circuit with $[n^\delta]$ inputs and n outputs so that the value of the ith output is equal to the value of the ith input if $i \geq n^\delta$ and 0 otherwise. Of course if the input is randomized uniformly, then we get a uniform distribution on the first $[n^\delta]$ outputs.

Now we define the circuit C that will give as an output the function F of a fooling pair.

Suppose that the circuits $D_0, \ldots, D_\mu$, $G_0, \ldots, G_\mu$ are pairwise disjoint. If $p_0, \ldots, p_\mu$, $q_0, \ldots, q_\mu$ are inputs for $D_0, \ldots, D_\mu$, $G_0, \ldots, G_\mu$ then $p = \langle p_0, \ldots, p_\mu, q_0, \ldots, q_\mu \rangle$ will be an input for C. C will have n outputs and the value of the *ith* output $C(p \mid i)$ is defined by the following boolean expressions. If $i \geq [n^\delta]$ then

$$C(p \mid i) = \bigvee_{j=0}^{\mu-1} [G_j(q_j \mid i) \wedge D_j(p_j \mid i) \wedge \bigwedge_{r=0}^{j-1} \neg D_r(p_r \mid i)] \qquad (*)$$

otherwise $C(p \mid i) = G_\mu(q_\mu \mid i)$.

[The meaning of $(*)$ is the following: if j is the smallest non-negative integer with $j < \mu$ and $D_j(p_j \mid i) = 1$ then $C(p \mid i) = G_j(q \mid i)$, if there is no such integer then $C(p \mid i) = 0$].

Clearly there is an (n^{c_5}, c_6) circuit with these properties where c_5, c_6 are absolute constants.

We define the parts of $P_0, \ldots, P_\mu$ in the partition as follows: thinking of $D_0, D_1, \ldots, D_{\mu-1}$ as subsets of A, we take P_j, $j < \mu$, to be all elements in D_j that do not belong to D_r, $r < j$, and P_μ are the remaining elements. Formally, for $j < \mu$

$$i \in P_j \leftrightarrow i \geqslant n^\delta \wedge D_j(p_j \mid i) = 1 \wedge \bigwedge_{r=0}^{j-1} D_r(p_r \mid i) = 1 \quad \text{if}$$

$$P_\mu = A - \bigcup_{j<\mu} P_j.$$

Now we prove that $\langle F, P \rangle$ is a fooling pair.

(1) follows immediately from the definition of F.
(2) The defining formula of P_j implies that the sets are disjoint and the definition of P_μ implies that they cover A.
(3) For $i < \mu$ $P_i = set(D_i) - \bigwedge_{j<i} set(D_j) - \{r \mid r < n^\delta\}$. The conditions $P_0 = A_0, \ldots, P_{i-1} = A_{i-1}$ restrict only the values of $D_0, \ldots, D_{i-1}$ but D_i is independent of them. Therefore $set(D_i)$ has the d-iterated small intersection property with parameters $(1-\delta, \delta/2, t)$ even with the conditions $P_0 = A_0, \ldots$. Since $P_i \subseteq set(D_i)$ it also has the same small intersection property.
(4) If $P_j = A_j$ then $F|_{A_j} = seq(G_j)|_{A_j}$. Since the outputs of each G_j are independent for $j = 0, \ldots, i$ the random variables $F|_{A_0}, \ldots, F|_{A_i}$ are also independent.
(5) If $P_i = A_i$ then $F|_{A_i} = seq(G_i)|_{A_i}$ and $seq(G_i)$ is (n,t)-uniform and as we have shown in the proof of (3) it is independent of P_i.
(6) Let $A_\mu = \{r \mid r < n^\delta\}$. $A_\mu \subseteq P_\mu$ always holds according to the definition of A_μ and P_μ. The definition of G_μ and $F|_{A_\mu} = seq(G_\mu)|_{A_\mu}$ implies that $F|_{A_\mu}$ has a uniform distribution so we have to prove only $Pr(P_\mu = A_\mu) \geqslant 1 - 2^{-n^\delta}$.

Let $r \in A - A_\mu$, $i < \mu$ be fixed. According to our assumption $Pr[r \in set(D_i)] \geqslant n^{-d\delta}$. Since the D_i's are independent we have $Pr[r \notin \bigcup_{i<\mu} set(D_i)] \leqslant (1 - n^{-d\delta})^\mu \leqslant (1 - n^{-\delta d})^{n^{(d+1)\delta}} \leqslant 2^{-n^\delta}$. This proves our assertion since $P_\mu \cap \bigcup_{i<\mu} set(D_i) = 0$.

The Main Theorem follows from Proposition 3 and Proposition 4 if we choose any $\delta > 0$ so that $\delta \to 0$ as $n \to \infty$.

5. DEPTH 2 CIRCUITS

The two results in this section are algorithms for the problem of approximate counting and finding a satisfying assignment, respectively, in depth 2 circuits (or CNF/DNF formulas). The two important parameters that affect the running time of the algorithms are the fraction of satisfying assignments and the sizes of clauses.

Let $A = \{x_1, \ldots, x_n\}$ be a set of boolean variables. A clause C is a conjunction of literals from A, e.g., $C = x_1 \wedge \bar{x}_3 \wedge x_6$. A DNF formula F is a disjunction of clauses (we will take F to be both the set of clauses and their disjunction, so $F = \bigvee_{C \in F} C$). $|F|$ denotes the number of clauses in F. For a clause or set of clauses H, $v(H)$ will denote the set of variables occurring in H. If for all $C \in F$, $|v(C)| \leqslant k$, then F is a $k-DNF$ formula. Similarly we define k-CNF formula.

We need some notation which is similar to that of Section 4.

Assume $A \subseteq v(F)$, and $Y \subseteq A$. We can restrict F by assigning values to variables in Y. If $y \in \{0,1\}^{|Y|}$, then $F_{Y,y}$ denotes the restricted formula after assigning y to Y (in order). Say that restrictions (Q, q) and (Y, y) satisfy $(Q, q) \geqslant (Y, y)$ if $Y \subseteq Q$ and q agrees with y on Y.

We further define F_Y to be the random variable (formula) $F_{Y,y}$ where $y \in \{0,1\}^{|Y|}$ is chosen uniformly at random. Note that if $Z \subseteq Y$ then $Pr(F_Y \not\equiv 0,1) \leqslant Pr(F_Z \not\equiv 0,1)$.

Theorem 1 deals with the approximation of depth 2 circuits. It shows that the output almost always depends on a small subset of the input variables.

THEOREM 1. Let C be a depth 2 circuit of bottom fan-in $\leqslant k$ with input variables in A. Then for every $r \geqslant 2^{2k^2}$ there exists a subset $Q \subseteq A$ s.t.

(1) $|Q| \leqslant r^{2k^2}$.

(2) $Pr(C_Q \not\equiv 0,1) \leqslant 2^{-r}$.

Furthermore, if $|A| = n$, the set Q can be found in time $O(n^{k^2})$.

Proof. It is enough to prove Theorem 1 in the case when the boolean formula F corresponding to C is a k-DNF formula. We prove the theorem by induction on k.

$k = 1$. If $|F| \leqslant r$ then set $G = F$ else let G be a subset of any r clauses in F. Let $Q = v(G)$. Then $|Q| \leqslant r \leqslant r^2$ and $Pr(F_Q \not\equiv 0,1) \leqslant 2^{-r}$.

$k > 1$. Assume the inductive assumption for all values less than k. Let G be a maximal subset of pairwise disjoint clauses from F formally $E, D \in G \to v(E) \cap v(D) = \varnothing$ but for all $E \in F - G$ there exist a $D \in G$ with $v(E) \cap v(D) = \varnothing$.

Case 1: $|G| > r2^k$. Let $\bar{G} \subsetneq G$ with $|\bar{G}| = r2^k$. Set $Q = v(\bar{G})$. We have $|Q| \leqslant kr2^k \leqslant r^{2k^2}$, and

$$Pr(F_Q \not\equiv 0, 1) \leqslant Pr(F_Q \not\equiv 1) \leqslant \prod_{E \in \bar{G}} Pr(E_Q \not\equiv 1) \leqslant (1 - 2^{-k})^{|\bar{G}|}$$
$$\leqslant (1 - 2^{-k})^{r2^k} \leqslant 2^{-r}.$$

Case 2: $|G| \leqslant r2^k$. Let $Z = v(G)$. Partition the clauses in $F - G$ into families, $H(Y, y)$ one for each $Y \subseteq Z$, $1 \leqslant |Y| \leqslant k - 1$, and $y \in \{0, 1\}^{|Y|}$, as follows: $H(Y, y) = \{E \in F - G | v(E) \cap Z = Y$ and $E_{Y,y} \not\equiv 0\}$. Clearly there are at most $(kr2^k)^k$ such families, and $F \equiv G \vee \bigvee_{Y,y} H(Y, y)$.

Consider the formulas $H_{Y,y} = H(Y, y)_{Y,y}$. $v(H_{Y,y}) \subseteq A - Z$, and each is a $(k-1) - DNF$ formula since G was maximal. Apply the inductive assumption to each $H_{Y,y}$ with parameters $k - 1$ for k and $2r$ for r. Let $Q_{Y,y}$ be the sets guaranteed inductively. Hence for all Y, y we have

(1) $|Q_{Y,y}| \leqslant (2r)^{2(k-1)^2}$ and
(2) $Pr[(H_{Y,y})_{Q_{Y,y}} \not\equiv 0, 1] \leqslant 2^{-2r}$.

Now set $Q = Z \cup \bigcup_{Y,y} Q_{Y,y}$. Then
(1) $|Q| \leqslant |Z| + \sum_{Y,y} |Q_{Y,y}| \leqslant rk2^k + (rk2^k)^k (2r)^{2(k-1)^2} \leqslant r^{2k^2}$.
To prove that $Pr(F_Q \not\equiv 0, 1) \leqslant 2^{-r}$ we observe the following. Let $q \in \{0, 1\}^Q$ s.t. $F_{Q,q} \not\equiv 0, 1$. Then $G_{Q,q} \equiv 0$, and in fact, $F_{Q,q} \equiv \bigvee_{(Q,q) > (Y,y)} (H_{Y,y})_{Q,q}$. Therefore for at least one pair (Y, y), $(H_{Y,y})_{Q,q} \not\equiv 0, 1$, and since $Q_{Y,y} \subseteq Q$ we have
(2) $Pr(F_Q \not\equiv 0, 1) \leqslant \sum_{Y,y} Pr[(H_{Y,y})_{Q,q} \not\equiv 0, 1] \leqslant (rk2^k)^k 2^{-2r} \leqslant 2^{-r}$.

The proof shows that the subset Q can be found in $DTIME(n^{k^2})$. [Indeed, let $h(k)$ denote the time necessary for finding Q. Since $|F| \leqslant n^k$, G can be found in time $O(n^k)$. $Q = Z \cup \bigcup_{Y,y} Q_{Y,y}$, which implies that Q can be found in time $O(n^k) + \sum_{Y,y} h(k-1) \leqslant n^k h(k-1)$ that is $h(k) \leqslant n^k h(k-1)$.]

Theorem 1 can be used as an algorithm for approximate counting (see Section 3.1).

Corollary 6. Let k, p, β be fixed, F any k-CNF or k-DNF formula, $p(F) = p$. Then the β-approximation problem can be solved in deterministic polynomial time.

Theorem 2 gives a somewhat faster algorithm for the simpler problem of finding a satisfying assignment.

THEOREM 2. Let F be a satisfiable k-CNF formula on n variables with $p = p(F)$. Then we can find a satisfying assignment of F in $DTIME(k|F| + 2^{k2^k(\log p^{-1})})$.

For example this theorem says that 3-CNF instances of SAT with a polynomially small fraction of satisfying assignments are easy, as we can find one in polynomial time!

Proof. (Sketch). Simple counting shows that any maximal set of clauses has at most $2^k(\log p^{-1})$ elements, otherwise some clauses would be satisfied. We try all assignments to this variable and proceed with induction on k.

REFERENCES

[Ad] L. Adleman, "Two theorems on random polynomial time," *19th FOCS* 75–83 (1978).

[Aj] M. Ajtai, "Σ_1^1-formula on finite structures," *Ann. Pure Appl. Logic* 24: 1–48 (1983).

[An] R. Anderson, "Set splitting," manuscrupt.

[AB] M. Ajtai and B. Ben-Or, "A theorem on probabilistic constant depth computations," *16th STOC* 471–474 (1984).

[ACGS] W. Alexi, B. Chor, O. Goldreich, and C. P. Schnorr, "RSA/Rabin bits are $1/2 + 1/poly(\log n)$ secure," *25th FOCS* 449–457 (1984).

[AW] M. Ajtai and A. Wigderson, "Deterministic simulation of probabilistic constant depth circuits," *26th FOCS* 11–19 (1985).

[BBS] L. Blum, M. Blum, and M. Shub, "A simple secure pseudo-random number generator," *Adv. Cryptogr. Proc. CRYPTO-82* 61–78 (1982).

[BM] M. Blum and S. Micali, "How to generate cryptographically strong sequences of pseudo random bits," *23rd FOCS* 112–117 (1982).

[ES] P. Erdős and J. Spencer, *Probabilistic Methods in Combinatorics*. Academic Press, New York, 1974.

[FLS] M. Furst, R. J. Lipton, and L. Stockmeyer, "Pseudo-random number generation and space complexity," *Information Control*, to appear.

[FSS] M. Furst, J. B. Saxe, and M. Sipser, "Parity circuits and the polynomial time hierarchy," *22nd* 260–270 (1981).

[Gi] J. Gill, "Complexity of probabilistic Turing machines," *SIAM J. Computing* 6: 675–695 (1977).

[GG] O. Gaber and Z. Galil, "Explicit construction of linear sized superconcentrators," *J Comp. Sys. Sci.* 22: 407–420 (1981).

[Ha] J. Hastad, "Lower bounds for the size of parity circuits," manuscript.

[KUW] R. M. Karp, E. Upfal, and A. Wigderson, "A fast parallel algorithm for the maximal independent set problem," *24th STOC* 266–272 (1984).

[KW] R. M. Karp and A. Wigderson, "A fast parallel algorithm for the maximal independent set problem," *24th STOC* 266–272 (1984).

[Lu] M. Luby, "A simple parallel algorithm for the maximal independent set problem," *17th STOC* (1985).

[Ma] G. A. Margulis, "Explicit construction of graphs without short cycles and low density codes," *Combinatorica* 2(1): 71–78 (1982).

[Ru] W. L. Ruzzo, "On uniform circuit complexity," *J. Comput. Sys. Sci.* 22(3): 365–383 (1981).

[RT] J. H. Reif and J. D. Tygar, "Towards a theory of parallel randomized computation," TR-07-84, Aiken Computation Lab., Harvard University, 1984.

[Si] M. Sipser, "A complexity theoretic approach to randomness," *15th STOC* 330–335 (1983).

[Sh] Shamir, "On the generation of crytographically strong pseudo random sequences," *8th ICALP*, Lecture notes in *Comp. Sci.* 62: 544–550. Springer-Verlag, Berlin, 1981.

[St] L. Stockmeyer, "The complexity of approximate counting," *15th STOC* 118–126 (1983).

[VV] L. G. Valiant and V. V. Vazirani, "NP is as easy as detecting unique solutions," *17th STOC* (1985).

[Ya] A. C. Yao, "Theory and applications of trapdoor functions," *23rd FOCS* 80–91 (1982).