2012 IEEE 27th Conference on Computational Complexity

DNF Sparsification and a Faster Deterministic Counting Algorithm

Parikshit Gopalan
Microsoft Research
Silicon Valley
Email: parik@microsoft.com

Raghu Meka
Institute for Advanced Study
Princeton
Email: raghu@ias.edu

Omer Reingold
Microsoft Research
Silicon Valley
Email: Omer.Reingold@microsoft.com

***Abstract*—We give a faster deterministic algorithm for approximately counting the number of satisfying solutions to a DNF or CNF. Given a DNF(or CNF) f on n variables and $\mathrm{poly}(n)$ terms, we give a deterministic $n^{\tilde{O}((\log\log n)^2)}$ time algorithm that computes an (additive) ε approximation to the fraction of satisfying assignments of f for $\varepsilon = 1/\mathrm{poly}(\log n)$. The previous best algorithm due to Luby and Velickovic from nearly two decades ago had a run-time of $n^{\exp(O(\sqrt{\log\log n}))}$.**

A crucial ingredient in our algorithm is a structural result which allows us to sparsify any small-width DNFformula. It says that any width w DNF(irrespective of the number of terms) can be ε-approximated by a width w DNFwith at most $(w\log(1/\varepsilon))^{O(w)}$ terms. Further, our approximating DNFs have an additional "sandwiching" property which is crucial for applications to derandomization. We believe the sparsification result to be of independent interest and use it to show a weak derandomization of the switching lemma wherein the random restrictions need only have limited independence.

I. Introduction

A natural way to present a Boolean function $f : \{0,1\}^n \to \{0,1\}$ is to write it as a CNF or DNF formula. The class of functions that admit compact representations of this form (aka polynomial size CNF and DNF formulae) are central to Boolean function analysis, computational complexity and machine learning.

Given a DNF formula f on n variables, the two natural size measures are the number of terms $m(f)$, and the max width of a term $w(f)$. The analogous measures for a CNF, are the number of clauses m and clause width w. It is folklore that every DNF formula f with m terms can be ε-approximated by another DNFg where $m(g) \leq m$ and $w(g) \leq \log(m/\varepsilon)$, regardless of $w(f)$. The formula g is a sparsification of f obtained by simply discarding all clauses of width larger than $\log(m/\varepsilon)$. In other words, *short* DNFformulas can be made *narrow*. An analogous statement can be derived for CNFs.

In this work, we show the reverse connection: *narrow* formulae can be made *short*. Indeed, we prove the existence of a strong form of approximation known as sandwiching approximations which are important in pseudorandomness (following [1]).

Theorem I.1. *For every* DNF *formula f with width w and every $\varepsilon > 0$, there exist* sandwiching DNF *formulae f_ℓ, f_u each with width w and at most $(w\log(1/\varepsilon))^{O(w)}$ terms such that*

$$f_\ell(x) \leq f(x) \leq f_u(x) \quad \forall x \in \{0,1\}^n,$$
$$\Pr_{x\in\{0,1\}^n}[f_\ell(x) \neq f_u(x)] \leq \varepsilon.$$

Our result is proved by a sparsification procedure for DNF formulae which uses the notion of quasi-sunflowers due to Rossman [2]. The best previously known result along these lines was due to Trevisan [3], who built on previous work by Ajtai and Wigderson [4]. Trevisan shows that every width w DNF can be ε-approximated (even in the sandwiching sense) by a decision tree of depth $d = O(w2^w \log(1/\varepsilon))$.

A corollary of our result is the following *junta theorem* for DNFs. A k-junta is a function which depends only on k variables.

Corollary I.2. *Every width-w* DNF *$f : \{0,1\}^n \to \{0,1\}$ is ε close to a $(w\log(1/\varepsilon))^{O(w)}$-junta.*

A similar but incomparable statement can be derived from Friedgut's junta theorem [5]. It is known that width w DNFs have average sensitivity at most w ([6]), so by Friedgut's theorem any width w DNF is ε-close to a $2^{\tilde{O}(w/\varepsilon)}$-junta. Friedgut's result gives better dependence on w, whereas we achieve much better dependence on ε. Fredigut's approximator $\tilde{f}$ is not *a priori* a small-width DNF, and one does not get sandwiching approximations.

1093-0159/12 $26.00 © 2012 IEEE
DOI 10.1109/CCC.2012.38

A. DNF *Counting and Pseduorandom Generators*

The problem of estimating the number of satisfying solutions to CNF and DNF formulae is closely tied to the problem of designing pseudorandom generators for such formulae with short seed-length. These problems have been studied extensively [7], [4], [8], [9], [10], [11], [12], [3], [1], [13], [14]. For a formula f, let $\mathsf{Bias}(f) = \mathsf{Pr}_{x \in_u \{0,1\}^n}[f(x) = 1]$. Given a formula f, the goal of a counting algorithm is to compute $\mathsf{Bias}(f)$. We refer to the counting problems for CNFs and DNFs as #CNF and #DNF respectively. The problem of computing $\mathsf{Bias}(f)$ exactly is #P-hard [15], hence we look to approximate $\mathsf{Bias}(f)$.

An algorithm gives an ε additive approximation for $\mathsf{Bias}(f)$ if its output is in the range $[\mathsf{Bias}(f) - \varepsilon, \mathsf{Bias}(f) + \varepsilon]$. It is easy to see that additive approximations for CNFs and DNFs are equivalent. There is a trivial solution based on random sampling, but finding a deterministic polynomial time algorithm has proved challenging.

Obtaining multiplicative approximations to $\mathsf{Bias}(f)$ is harder, and here the complexities of #CNF and #DNF are very different. An algorithm is said to be a c-approximation algorithm if its output lies in the range $[\mathsf{Bias}(f), c\mathsf{Bias}(f)]$. It is easy to see that obtaining a multliplicative approximation for #CNF is NP-hard. Karp and Luby gave the first multiplicative approximation for #DNF. They also showed the following reduction between additive and multiplicative approximations for #DNF: for DNF formulae with m terms, a deterministic (randomized) $\frac{\varepsilon}{m}$ additive approximation to #DNF can be used to give a deterministic (randomized) $(1+\varepsilon)$ multiplicative approximation.

Derandomizing thes Karp-Luby algorithm is an important problem in derandomization that has received a lot of attention starting form the work of Ajtai and Wigderson [4], [16], [10], [12], [11], [3]. The best previous result is due to Luby-Velickovic [10], [11] from nearly two decades ago: they gave a deterministic $n^{\exp(O(\sqrt{\log \log n}))}$ time algorithm that can compute an ε-additive approximation for any fixed constant ε [10], [11].

A natural approach to this problem is to design pseudorandom generators (PRGs) with small seeds that can ε fool depth two circuits. This problem and its generalization to constant depth circuits are central problems in pseudorandomness [1], [3], [8], [9], [11]–[14], [17] .

Definition I.3. *A generator $G : \{0,1\}^r \to \{0,1\}^n$ ε-fools a class $\mathcal{F}$ of functions if*

$$|\mathop{\mathsf{E}}_{y \in_u \{0,1\}^r}[f(G(y))] - \mathsf{Bias}(f)| < \varepsilon$$

for all $f \in \mathcal{F}$. The function G should be polynomial time computable.

A generator with seed-length r that ε-fools DNFs with m clauses gives a an ε-additive approximation for $\mathsf{Bias}(f)$ in $\mathrm{poly}(m, n, 2^r)$ time by enumerating over all seeds. Such an algorithm only requires black-box access to f. The Krap-Luby reduction implies that an optimal psedurandom generator for DNFs with seedlength $O(\log(mn/\varepsilon))$ will give a deterministic multiplicative approximation algorithm for #DNF. The Luby-Velikovic algorithm is a not a black-box algorithm, but PRGs for small-width DNFs are an important ingredient.

Our Results: We use our sprasification lemma to give a better PRG for the class of width w DNFformulae on n variables, which we denote by $\mathsf{DNF}(w, n)$.

Theorem I.4. *For all $\varepsilon, w > 0$, there is a generator that ε-fools $\mathsf{DNF}(w, n)$ and has seed-length*

$$r = \tilde{O}(w^2 + \log^2(1/\varepsilon) + \log \log n).$$

In comparison, Luby and Velickovic [11] give a PRG with seed-length $O(2^w + \log \log n)$ for fooling width w DNFs. Note that for $w = O(\log \log n)$, the seed-length of the our generator is $\tilde{O}((\log \log n)^2)$, whereas Luby and Velickovic need seed-length $\log^{O(1)} n$.

The generator for small-width DNFs is a crucial ingredient in the Luby-Velickovic counting algorithm. Given our improvement, we can improve and simplify their analysis to get a faster deterministic counting algorithm. This is the first progress on this well-studied problem in a while. In addition, we can allow for smaller values of ε. The dependence on ε is important since to get a multiplicative approximation using the Karp-Luby reduciton, one requires ε to be polynomially small.

Theorem I.5. *Given any polynomial size $\mathsf{DNF} f : \{0,1\}^n \to \{0,1\}$, there is a deterministic algorithm running in time $n^{\tilde{O}((\log \log n)^2)}$ that computes an ε-additive approximation to $\mathsf{Bias}(f)$ for $\varepsilon = 1/(\log n)^{O(1)}$.*

As an additional application of our sparsification result, we give a partial derandomization of the switching lemma. The parameters we obtain are close to that of the previous best results due to Ajtai and Wigderson [4] and perhaps more importantly, our argument is conceptually much simpler involving iterative applications of our sparsification result and a naive union bound. We defer the details to Section II-C.

II. Outline of Results

We first describe the proof of our sparsification result as it plays a crucial role in our final algorithm for approximating the number of solutions to DNFs.

A. DNF *Sparsification*

For intuition, we first describe how to get a weaker version of Theorem I.1 with a bound of $(2^w \log(m/\varepsilon))^{O(w)}$.

Fix a width w DNF $f = T_1 \vee T_2 \vee \cdots \vee T_m : \{0,1\}^n \to \{0,1\}$ and for simplicity suppose that f is monotone; the arguments work almost unchanged for the general case. The starting point of our sparsification result is the idea of using the Erdős-Rado *sunflower lemma* in the context of DNFs originally due to Luby and Velickovic [11]. If $m > w!(k-1)^w$ for some k, then by the Erdős-Rado sunflower lemma, f has a *sunflower* of size k. Here, a sunflower is a collection of terms $T_{i_1}, \ldots, T_{i_k}$ such that for $T = \cap_{j=1}^{k} T_{i_j}$, the sets $T_{i_j} \setminus T$ are disjoint (we let T_i denote the set of variables occurring in T_i as well).

Now, consider the formula f' obtained from f by replacing the terms $T_{i_1}, T_{i_2}, \cdots, T_{i_k}$ with T. Then, $f(x) \leq f'(x)$. Further, $f'(x) = 1$ and $f(x) = 0$ only if the assignment x does not satisfy any of the terms $T_{i_j} \setminus T$ for $j \in [k]$. Therefore, for $x \in_u \{0,1\}^n$,

$$\mathsf{Pr}[f'(x) \neq f(x)] \leq$$
$$\mathsf{Pr}[x \text{ does not satisfy } \vee_{j=1}^{k} (T_{i_j} \setminus T)] \leq \left(1 - \frac{1}{2^w}\right)^k,$$

where the last inequality follows from the fact that the terms $T_{i_j} \setminus T$ are disjoint and each has width at most w. Thus, if $k = 2^w \ln(mw/\varepsilon)$, the error in approximating f by f' is at most ε/mw. We can now iteratively apply the above argument as long as the number of terms is larger than $w!(k-1)^w$ and in each application we simplify the formula by shrinking at least one of the terms. Thus, we only need repeat the process at most mw times, obtaining an *upper* approximating formula f_u with the promised bound on the number of terms, $f \leq f_u$ and error $\mathsf{Bias}(f_u) - \mathsf{Bias}(f) \leq (mw) \cdot (\varepsilon/mw) = \varepsilon$.

We next describe the construction of the *lower* approximating formula f_ℓ. We start by finding a sunflower $T_{i_1}, \cdots, T_{i_k}$ as in the construction of f_u. Now consider a formula f'' obtained from f by dropping one of the terms, say T_{i_1}. Then, $f''(x) \leq f(x)$. Further, by an argument similar to the above,

$$\mathsf{Pr}[f''(x) \neq f(x)] \leq$$
$$\mathsf{Pr}[x \text{ does not satisfy } \vee_{j=2}^{k} (T_{i_j} \setminus T)] \leq \left(1 - \frac{1}{2^w}\right)^{k-1} < 2\varepsilon/mw.$$

We now iteratively apply the above step to obtain a formula f_ℓ with the promised number of terms, $f_\ell \leq f$ and $\mathsf{Bias}(f) - \mathsf{Bias}(f_\ell) \leq m(2\varepsilon/mw) \leq \varepsilon$.

Note that the above argument already beats the previous best width vs size trade-off results which needed approximations by $2^{2^{O(w)}}$ term DNFs [3], [4]. (Strictly speaking, we have a $(\log m)^{O(w)}$ dependence on m which could be worse. However, the dependence on m can be eliminated by a more careful calculation of the error and we ignore this issue in this informal description). To prove the stronger version of Theorem I.1, observe that the main property of the sunflower system we used above is that the formula $(T_{i_1} \setminus T) \vee (T_2 \setminus T) \vee \cdots \vee (T_{i_k} \setminus T)$ is highly biased towards 1. As shown by Rossman [2], we can guarantee the existence of such "quasi-sunflower" systems satisfying the above weaker property, even when the number of terms is much smaller than in the usual sunflower lemma. We then carefully adapt our argument to use quasi-sunflowers instead of sunflowers, to obtain Theorem I.1.

B. The Deterministic Counting Algorithm

As a first step towards our final counting algorithm for DNFs, we give a pseudorandom generator (PRG) that δ-fools width w DNFs on n variables with seed-length $\tilde{O}(w^2 + \log^2(1/\delta)) + O(\log \log n)$. Our seed-length is exponentially better than that of Luby and Velickovic in terms of width and this improvement turns out to be critical in improving the counting algorithm for general DNFs.

The improved generator for small-width DNFs works as follows: We first use our sparsification result to reduce the case of fooling width w DNFs with an arbitrary number of terms to that of fooling width w DNFs with $2^{\tilde{O}(w)}$ terms and then apply recent results (Theorem IV.1, [14]) showing that small-bias spaces with $\exp(-\tilde{O}(\log^2 m))$-bias fool DNFs with m terms. The fact that our sparsification in fact gives sandwiching approximators is critical for this step.

We are now ready to present our deterministic algorithm for approximating the number of satisfying solutions to a DNF. The high level outline of the algorithm is similar to that of Luby and Velickovic and is as follows.

Let $f = T_1 \vee T_2 \vee \cdots \vee T_m$ be a DNFon n variables with $m = \text{poly}(n)$. Without loss of generality suppose that f has width at most $O(\log n)$.

Let $\mathcal{H} : [n] \to [t]$ be a family of $O(\log \log n)$-wise independent hash functions for $t = O(\log n)$. We choose a hash function $h \in_u \mathcal{H}$ and partition $[n]$ into t "buckets" by setting $B_j = \{i : h(i) = j\}$ for $j \in [t]$. Then, for any $i \in [m]$ with probability at least $1 - 1/\text{poly}(\log n)$, the term T_i will be split almost evenly across the buckets so that $\max_{j \in [t]} |B_j \cap T_i| < w = O(\log \log n)$. Call a term T_i *bad* if the above property is not satisfied.

Let f' denote the DNFobtained by dropping all the bad terms. We now wish to exploit the fact that when restricted to the coordinates within a single bucket B_j, $f'_{|B_j}$ is a width at most w DNF. So we can use our generator for small-width DNFs to fool $f'_{|B_j}$ for any arbitrary fixing of the variables in the remaining buckets. We now use a hybrid argument to show that f' is fooled by the generator which uses independent copies of the generator from Theorem IV.1 within each of the t buckets $B_1, B_2, \ldots, B_t$. The seed-length of the final generator for fooling f' is $t \cdot \tilde{O}(w^2 + \log \log n) = \tilde{O}(\log n)$. Thus, we can in particular estimate the bias of f' deterministically within an additive error of $1/\text{poly}(\log n)$ in time $\exp(\tilde{O}(\log n))$.

Finally, we need to relate the bias of f' with that of f. This simply follows from the fact that each term T_i of f is not in f' with a probability of at most $1/\text{poly}(\log n)$. Combining the above arguments and setting the parameters appropriately, we get a deterministic $\exp(\tilde{O}(\log n))$ time algorithm for approximating the bias of f within an additive $1/\text{poly}(\log n)$ error.

C. Derandomizing the Switching Lemma

The switching lemma [18] is one of the central techniques in proving lowerbounds for small-depth circuits and says the following.

Given $L \subseteq [n]$ and $x \in \{0,1\}^{[n] \setminus L}$ define a *restriction* $\rho \equiv \rho_{L,x} \in \{*, 0, 1\}^n$ by $\rho_i = *$ if $i \in L$ and $\rho_i = x_i$ otherwise. We call the set $L \equiv L(\rho)$ as the set of "live" variables. For $f : \{0,1\}^n \to \{0,1\}$, and $\rho \in \{*, 0, 1\}^n$, define $f_\rho : \{0,1\}^{L(\rho)} \to \{0,1\}$ by $f_\rho(y) = f(x)$, where $x_i = y_i$ for $i \in L(\rho)$ and $x_i = \rho_i$ otherwise. Given a distribution $\mathcal{D}$ on $2^{[n]}$, let $\mathcal{D}$ (abusing notation, the meaning will be clear from context) denote the distribution on $\rho \in \{*, 0, 1\}^n$ by setting $\rho = \rho_{L,x}$ where $L \leftarrow \mathcal{D}$ and $x \in_u \{0,1\}^{[n] \setminus L}$. Finally, for $p \in (0, 1)$, let $\mathcal{D}_p$ denote the distribution on subsets L of $[n]$ where each element $i \in [n]$ is present in L independently with probability p.

Theorem II.1 (Switching Lemma, [18]). *Let $f : \{0,1\}^n \to \{0,1\}$ be a DNF of width w and let $\rho \leftarrow \mathcal{D}_p$. Then, $\Pr[f_\rho$ cannot be computed by a decision tree of depth $s]$ $< (5pw)^s$.*

Given the fundamental nature of the switching lemma it is natural to ask if there exists a *derandomized* version of the switching lemma in the sense of choosing the set of live variables in a pseudorandom way. One could even ask for a stronger derandomization where the assignments to the non-live variables are also chosen pseudorandomly, however, this is unnecessary for most interesting applications and we limit ourselves to the former case here.

The only result we are aware of in this direction is that of Ajtai and Wigderson [4].

Theorem II.2 ([4]). *For all $\gamma > 0$, $p < 1/n^\gamma$, there is a distribution $\mathcal{D}$ on $\binom{[n]}{pn}$ with $L \leftarrow \mathcal{D}$ samplable using $O_\gamma(\log n)$ random bits such that the following holds. For $\rho \leftarrow \mathcal{D}$, and any polynomial size DNFf, $\Pr[f_\rho$ depends on more than $O_\gamma(1)$ variables$]$ $< 1/\text{poly}(n)$.*

Here, we give a different argument that essentially recovers the result of Ajtai and Wigderson and further gives a trade-off between the survival probability p, the complexity of the restricted function and the failure probability of the restriction. Our argument is much simpler than those of Hastad and Ajtai and Wigderson involving iterative applications of Theorem I.1.

Theorem II.3. *There exists a constant C such that for any $w, s, \delta > 0$ and $p < \delta/(w \log(1/\varepsilon))^{C \log w}$, there is a distribution $\mathcal{D}$ on $2^{[n]}$ such that $L \leftarrow \mathcal{D}$ can be sampled using*

$$r(n, s, \varepsilon, \delta) = O\left((\log w) \cdot (\log n + s \log(1/\delta)) + w \log(w \log(1/\varepsilon))\right)$$

random bits, the indicator events $1\{i \in L\}$ are p-biased and the following holds. For any width w DNF$f : \{0,1\}^n \to \{0,1\}$, and $\rho \leftarrow \mathcal{D}$, $\Pr[f_\rho$ is not ε-strongly approximable by width s DNF$s]$ $< \varepsilon + \delta^{s/4}$.

In particular, by setting $\delta = 1/n^\gamma$, $s = \Theta(1/\gamma)$, $\varepsilon = 1/\text{poly}(n)$, $w = O(\log n)$, we almost recover the derandomized switching lemma of Ajtai and Wigderson, with the main difference being that we need $O((\log n)(\log \log n))$ bits to sample from $\mathcal{D}$ and we only get f_ρ is strongly approximable by width $O_\gamma(1)$ DNFs.

We remark that if instead of finding a small set of restrictions that work for all formulas f, we are given the formula f as input, Agrawal et al. [19] give a polynomial-time algorithm to find a restriction that simplifies the formula as well as the bounds given by the switching lemma Theorem II.1.

Our derandomization is based on the intuition that the switching lemma is easier to show when the number of terms in the original DNFf is small in terms of the width w of f. For example, let $f = T_1 \vee T_2 \vee \cdots \vee T_{2^w}$ be a width w DNF. Note that for $0 < p < 1$, and $\rho \leftarrow \mathcal{D}_p$, the probability that a single term T_i survives the restriction f_ρ (is not set to be a constant) is at most

$$\sum_{i=1}^{w} \binom{w}{i} p^i ((1-p)/2)^{w-i} < \left(\frac{1+p}{2}\right)^w.$$

In particular if $p < 1/w$, the above probability is at most $e/2^w$. Thus, by linearity of expectation, the expected number of terms that survive the restriction is at most $O(1)$. Hence, by Markov's inequality, the restricted DNFf_ρ has very few surviving terms with high probability. Further, as we are only using Markov's inequality, the above argument would work even if the restriction ρ is sampled from a distribution where the choices for different variables are only k-wise independent for $k > w$.

The idea now is to use our sparsification result Theorem I.1 to first reduce the case of arbitrary DNFs of small-width to that of DNFs with a small number of terms and then use an argument similar to the above. Unfortunately, the bound in Theorem I.1 is not enough to directly yield a strong bound on the expected number of surviving terms (and such a strong bound is not possible in general). However, there is a lot of slack in the above argument, as for instance even if the number of surviving terms is large, most of them have much smaller width in the restriction. We exploit this fact as follows.

Observe that for any $r \geq 1$, a random restriction f_ρ with probability $p = w^{-O(r)}$ of being alive, can be viewed as the result of a sequence of r random restrictions each with a $1/w^{O(1)}$ fraction of live variables. We use the latter view and argue that in each round, when we take a random restriction with $1/w$ fraction of variables surviving, the width of the formula decreases by half with high probability and then iteratively apply the argument to the new width $w/2$ formulas. The number of rounds needed to simplify the formula to constant width would be roughly $O(\log w)$ and this corresponds to a random restriction where the probability of being alive is $\exp(-\Omega(\log^2 w))$. Moreover, throughout our arguments only rely on upper bounds on the probability that a set of at most w variables remain alive and hence carry over straightforwardly to the case where the random restrictions only have limited independence, yielding Theorem II.3.

III. DNF Sparsification

We shall use an extension of the classical sunflower lemma of Erdős and Rado due to Rossman [2].

Call a DNF f *unate* if no variable has both itself and its negation appearing in the formula.

Lemma III.1 (Quasi-Sunflower Lemma, [2])**.** *For any unate DNF $f \equiv T_1 \vee \cdots \vee T_m$ of width at most w, there exists a collection of terms $T_{i_1}, \ldots, T_{i_s}$ such that for $Y = \cap_{j=1}^{s} T_{i_j}$,*

$$\Pr_{x \in_u \{0,1\}^n}[x \text{ satisfies } (T_{i_1} \setminus Y) \vee (T_{i_2} \setminus Y) \cdots \vee (T_{i_s} \setminus Y)] > 1 - \exp\left(\frac{-(m/w!)^{1/w}}{2.47}\right).$$

Rossman states the result in the language of set systems, which we have rephrased in the language of DNFs. We show the equivalence of the two in the appendix. As mentioned in the introduction, the above lemma is an extension of the classical sunflower lemma in the sense that even though the "petals" $(T_{i_j} \setminus Y)$ are not necessarily disjoint, we have a guarantee that the probability that none of them is satisfied is small.

We also need the following simple facts.

Fact III.2. *For any width w DNF $f = T_1 \vee T_2 \vee \cdots \vee T_m$ with no clause containing a variable and its negation, there exists a sub-formula $f' = T_{j_1} \vee \cdots \vee T_{j_s}$ that is unate and $s \geq m/2^w$.*

Proof: Pick a random set of literals S as follows: for each of the variables x_i add one of x_i or $\bar{x}_i$ to S uniformly at random. Let f_S be the sub-formula of f formed of terms containing only literals from S. Then, f_S is always unate and by linearity of expectation $\mathbb{E}[\text{Size of } f_S] \geq m/2^w$. ∎

Fact III.3. *If $f = T_1 \vee T_2 \vee \cdots \vee T_s$ is a unate DNF, then,*

$$\Pr[T_2 \vee T_3 \vee \cdots \vee T_s = 0] \leq 2^{|T_1|} \Pr[f = 0].$$

Proof: Without loss of generality suppose that f is monotone. Let $f' = T_2 \vee T_3 \cdots \vee T_s$. Then, $\Pr[f = 0] = \Pr[T_1 = 0] \cdot \Pr[f' = 0 | T_1 = 0]$. Now, as f is monotone,

$\mathsf{Pr}[f' = 0 | T_1 = 0] \geq \mathsf{Pr}[f' = 0]$. The claim now follows. ■

We can now prove the main structural result.

Proof of Theorem I.1: Let $f = T_1 \vee T_2 \vee \cdots \vee T_m$. Without loss of generality assume that no term T_i contains a variable and its negation. Let f_1 be a unate subformula of f with at least $m/2^w$ clauses as guaranteed by Fact III.2. Let $c = 2.47$ be the constant in Lemma III.1. Further, assume that $\varepsilon < 1/6$.

By Lemma III.1, there exists a collection of terms $T_{i_1}, \ldots, T_{i_s}$ from f_1 and a subset of literals Y that does not contain both a variable and its negation such that for $x \in_u \{0,1\}^n$,

$$\mathsf{Pr}[x \text{ satisfies } (T_{i_1} \setminus Y) \vee \cdots (T_{i_s} \setminus Y)] > 1 - \exp\left(\frac{-(m/2^w w!)^{1/w}}{c}\right) \equiv 1 - \delta_w(m). \quad \text{(III.1)}$$

We first show the existence of a smaller lower-sandwiching DNF. Let f' be the formula obtained from f by dropping the term T_{i_1}. Then, if an assignment satisfies f' it also satisfies f. Hence, $f'(x) \leq f(x)$ for all x. Furthermore, by Equation III.1, and Fact III.3,

$$\mathsf{Pr}_{x \in_u \{0,1\}^n}[f(x) = 1 \wedge f'(x) = 0] \leq \mathsf{Pr}[x \text{ does not satisfy } (T_{i_2} \setminus Y) \vee \cdots \vee (T_{i_s} \setminus Y)] \leq 2^w \mathsf{Pr}[x \text{ does not satisfy } (T_{i_1} \setminus Y) \vee \cdots \vee (T_{i_s} \setminus Y)] < 2^w \delta_w(m).$$

Recall that $W = w^{3w}(50 \log(1/\varepsilon))^w$. We now iterate the argument untill we get a formula f_ℓ with at most W terms. The total error will be (we defer the tedious calculation to the appendix)

$$\mathbb{E}[f(x)] - \mathbb{E}[f_\ell(x)] < \sum_{j=W+1}^{m} 2^w \delta_w(j) < \varepsilon. \quad \text{(III.2)}$$

We next show the existence of the upper sandwiching DNF f_u. Find terms $T_{i_1}, \ldots, T_{i_s}$ as in Equation III.1. Let $1 \leq j \leq s$ be such that $Y \subsetneq T_{i_j}$. Let f'' be the formula obtained from f by replacing the term T_{i_j} with Y (if Y is empty we use constant 1). Note that if an assignment satisfies f, then it also satisfies f''. Thus, $f(x) \leq f''(x)$ for all x. On the other hand, if $f''(x) = 1$, then for $f(x)$ to be 0, x must not satisfy any of the terms $T_{i_1} \setminus Y, T_{i_2} \setminus Y, \cdots, T_{i_s} \setminus Y$. Therefore, by Equation III.1,

$$\mathsf{Pr}_{x \in_u \{0,1\}^n}[f(x) = 0 \wedge f''(x) = 1] \leq \delta_w(m).$$

In particular $\mathbb{E}[f(x)] - \mathbb{E}[f''(x)] \leq \delta_w(m)$. We can now apply the argument iteratively until we get a formula f_u with at most W terms. Every time we apply the argument, we reduce the width of at least one of the terms by at least one. Therefore, the total error incurred will be at most (we again defer the calculation to the appendix):

$$\mathbb{E}[f_u(x)] - \mathbb{E}[f(x)] < \sum_{j=W+1}^{m} (jw)\delta_w(j) < \varepsilon. \quad \text{(III.3)}$$

A similar argument shows the lemma for CNFs. ■

IV. Fooling Small-Width DNFs

We next use our sparsification result to construct a pseudorandom generator for small-width DNFs, obtaining an exponential improvement in terms of the width over the generator of Luby and Velickovic [11].

We say a generator $G : \{0,1\}^r \to \{0,1\}^n$ δ-fools a class of functions $\mathcal{F}$ if $|Bias(f) - Bias(f \circ G)| < \delta$. We refer to r as the seed-length of G and say the generator G is explicit if there is a polynomial (in n) time algorithm that can compute G.

Theorem IV.1. *For all $\delta, w > 0$, there exists an explicit generator $G : \{0,1\}^r \to \{0,1\}^n$ that δ-fools width w DNFs on n variables and has seed-length*

$$r = O\left(\left(w^2 + \log^2(1/\delta)\right) \cdot \left(\log^3 w + \log^2 \log(1/\delta)\right) + \log \log n\right) = \tilde{O}\left(w^2 + \log^2(1/\delta)\right) + O(\log \log n).$$

We prove the theorem as follows: We first use our sparsification result to reduce the case of fooling width w DNFs with an arbitrary number of terms to that of fooling width w DNFs with $2^{\tilde{O}(w)}$ terms and then apply the recent results showing that small-bias spaces with $\exp(-\tilde{O}(\log^2 m))$-bias fool DNFs with m terms.

Definition IV.2 (k-wise ε-biased spaces)**.** *A distribution $\mathcal{D}$ over $\{0,1\}^n$ is said to be (k,ε)-biased space if for every set $I \subseteq [n]$, $1 \leq |I| \leq k$, $|\mathsf{Pr}_{x \leftarrow \mathcal{D}}[\oplus_{i \in I} x_i = 1] - 1/2| < \varepsilon$.*

There exist explicit (k,ε)-biased spaces that require only $O(k + \log(1/\varepsilon) + \log \log n)$ bits to sample from – see [20].

Finally, we need the following result of De et al. [14] showing that small-biased spaces fool DNFs.

Theorem IV.3. *There exists a constant C such that for every $\delta > 0$ a DNF $f : \{0,1\}^n \to \{0,1\}$ with m terms is δ-fooled by (k,ε)-biased distributions for $k = O(\log^2(m/\delta))$, $\varepsilon = (\log(m/\delta))^{-C(\log(m/\delta))^2}$.*

De et al. prove the above statement only for the case of $k = n$, however, their argument extends straightforwardly to the above case as well. (De et al. prove the result by constructing *small* ℓ_1*-norm* sandwiching approximators, which also happen to be of degree at most $O(\log^2(m/\delta))$).

We are now ready to prove the main result of this section. We use the following simple fact.

Fact IV.4. *Let $f_\ell, f, f_u : \{0,1\}^n \to \{0,1\}$ be such that $f_\ell(z) \le f(z) \le f_u(z)$ for all z. Then, for any two distributions $\mathcal{D}_1, \mathcal{D}_2$ on $\{0,1\}^n$, and $x \leftarrow \mathcal{D}_1$, $y \leftarrow \mathcal{D}_2$,*

$$|\mathbb{E}[f(x)] - \mathbb{E}[f(y)]| < |\mathbb{E}[f_\ell(x)] - \mathbb{E}[f_\ell(y)]| + |\mathbb{E}[f_u(x)] - \mathbb{E}[f_u(y)]| + |\mathbb{E}[f_u(x)] - \mathbb{E}[f_\ell(x)]|.$$

Proof of Theorem IV.1: We claim that (k,ε)-biased spaces fool width w DNFs for $k = c(w^2 + \log^2(1/\delta))(\log^2 w + \log^2\log(1/\delta)))$ and $\varepsilon = k^{-Ck^2}$ for c, C sufficiently large constants. Let f be a width w DNF. Let f_ℓ, f_u be width w DNFs with at most $w^{3w}(C\log(1/\delta))^w$ terms that δ-strongly approximates f as in Theorem I.1. Let $\mathcal{D}$ be a (k,ε)-biased space for c, C sufficiently large so that by Theorem IV.3, for $x \in_u \{0,1\}^n$ and $y \leftarrow \mathcal{D}$,

$$|\mathbb{E}[f_u(x)] - \mathbb{E}[f_\ell(x)]| < \delta, \quad |\mathbb{E}[f_\ell(x)] - \mathbb{E}[f_\ell(y)]| < \delta,$$

$$|\mathbb{E}[f_u(x)] - \mathbb{E}[f_u(y)]| < \delta.$$

The theorem now follows from Fact IV.4 applied to random variables x, y. ∎

V. Deterministic Counting for DNFs

We now plug the better PRG for small-width DNFs from the previous section into the Luby-Velickovic counting algorithm [11]. The better seed-length means that we do not need to balance various parameters as carefully, and can redo their arguments with simpler and better settings of parameters.

We need the following lemma about k-wise independent hash functions.

Lemma V.1. *Let $\mathcal{H} : [n] \to [t]$ be a k-wise independent family of hash functions. Then, for every set $S \subseteq [n]$, $|S| \le kt$,*

$$\Pr_{h \in_u \mathcal{H}}[\,|h^{-1}(1) \cap S| \ge 6k\,] \le 2^{-k}.$$

Proof: Without loss of generality suppose that $S = \{1, \ldots, kt\}$. Let $X_1, \ldots, X_{kt}$ be indicator random variables that are 1 if $h(i) = 1$ and 0 otherwise. Then,

$$\Pr_{h \in_u \mathcal{H}}[\,|h^{-1}(1) \cap S| \ge 6k\,] \cdot \binom{6k}{k} \le \mathbb{E}\left[\sum_{I \subseteq [kt], |I| = k} \prod_{i \in I} X_i\right] \le \binom{kt}{k} \cdot \frac{1}{t^k} \le e^k.$$

The lemma now follows. ∎

Theorem V.2 (Main)**.** *There exists a deterministic algorithm that given a width W DNF formula $f : \{0,1\}^n \to \{0,1\}$ with m terms computes a number p such that $|p - Bias(f)| < \varepsilon$ in time $O(m \cdot n^{O(\log(W/\varepsilon))} \cdot \exp(W \cdot \tilde{O}(\log^2(W/\varepsilon) + \log\log n)))$. In particular, for any DNF of polynomial size, the run time is $n^{\tilde{O}((\log\log n)^2)}$ to get error $\varepsilon = 1/\mathrm{poly}(\log n)$.*

Proof: The second part follows immediately from the first as for the purpose of approximating the bias within error ε, any DNF with $m = \mathrm{poly}(n)$ terms can be assumed to have width at most $\log(m/\varepsilon) = O(\log(n/\varepsilon))$. This is because we can truncate any terms with more than $\log(m/\varepsilon)$ variables to $\log(m/\varepsilon)$ variables with only a loss of ε/m in the bias. We now show the first part.

Let $\mathcal{H} = \{h : [n] \to [t]\}$ be a family of k-wise independent hash functions for $k = \log(W/\varepsilon)$ and $t = W$. Let $f = T_1 \vee T_2 \vee \cdots \vee T_m$.

Pick a random hash function $h \in_u \mathcal{H}$ and let $B_j = \{i : h(i) = j\}$. Call a term T_i *bad* if $\max_{j \in [t]} |B_j \cap S| > \log(W/\varepsilon)$. Let f_h be the formula obtained from f by dropping all the bad terms. We claim that the bias of f and bias of f_h are close most of the time.

Claim V.3. $|Bias(f) - \mathbb{E}_h[Bias(f_h)]| < \varepsilon$.

Proof: As f_h is obtained by dropping a few terms in f, $\Pr[f_h(x) = 1] \le \Pr[f(x) = 1]$ for every hash function h. Further, if $x \in \{0,1\}^n$ is a satisfying assignment for f, then x satisfies at least one of the terms, say T_i, of f. Now, by Lemma V.1 and a union bound, the probability that the term T_i is not included in f_h is at most ε. Therefore every satisfying assignment is a satisfying assignment for f_h with probability at least $1 - \varepsilon$. The claim now follows. ∎

Fix the choice of the hash function $h \in \mathcal{H}$. We next show how to approximate the bias of f_h. Let $w = \log(W/\varepsilon)$. Let G_w be the generator from Theorem IV.1 that fools width w DNFs with error at most $\delta = \varepsilon/W$ and seed-length r_w as in the theorem.

Define a generator $G : (\{0,1\}^{r_w})^W \to \{0,1\}^n$ as

follows:

$$G(z_1,\ldots,z_W) = x, \text{ where for } j \in [W],\ x_{|B_j} = G_w(z_j). \tag{V.1}$$

That is, G takes an independent copy of G_w in each of the W buckets $B_1,\ldots,B_W$. We claim that G fools f_h with error at most ε. To see this, first observe that for any $j \in [W]$ and for any arbitrary fixing of the variables $x_i, i \notin B_j$, the formula f_h restricted to the variables in B_j is a DNF with width at most w. Thus, the restriction of f_h to variables in B_j is fooled by the generator G_w with error at most δ by Theorem IV.1. For $j \in [W]$ let $\mathcal{D}_0$ be the uniform distribution over $\{0,1\}^n$ and let $\mathcal{D}_j$ be the distribution obtained from $\mathcal{D}_j$ by replacing the variables in bucket B_j with an independent copy of the generator G_w. Then, the above argument shows that for $j \geq 1$,

$$|\Pr_{x\leftarrow\mathcal{D}_{j-1}}[f_h(x)=1] - \Pr_{x\leftarrow\mathcal{D}_j}[f_h(x)=1]| < \delta.$$

As $\mathcal{D}_W$ is the same distribution as the output of G, it follows from an iterative application of the above equation that G fools f_h with error at most $W\delta = \varepsilon$. In particular, we can estimate $Bias(f_h)$ within error ε by evaluating f_h at the support of the output of G and computing the fraction of satisfying assignments. Thus, evaluating f_h at the support of G for all $h \in \mathcal{H}$ gives us the desired estimate for the bias of f by Claim V.3. The bound on the runtime follows from the fact that G has support $\exp(W \cdot \tilde{O}(\log^2(W/\varepsilon) + \log\log n))$, $|\mathcal{H}| = n^{O(\log(W/\varepsilon)}$ and we only need $O(m)$ time per each evaluation of f_h at an assignment. ■

VI. A Derandomization of Switching Lemma

For $k \leq n$, let $\mathcal{D}_p(k)$ denote the class of distributions on subsets $L \subseteq [n]$ where the indicator random variables $1\{i \in L\}$ are p-biased and for any $I \subseteq [n], |I| \leq k$, $\Pr[I \subseteq L] \leq 2p^{|I|}$. Note that there exist explicit distributions $\mathcal{D} \in \mathcal{D}_p(k)$ that can be sampled using $O(k\log(1/p) + \log n)$-random bits (for instance, one can use p^k-almost k-wise independent p-biased variables such as those constructed in [20]).

We first prove Theorem II.3 assuming the following claim.

Claim VI.1. *There exists a constant $c < 1$ such that the following holds for all $\delta, \varepsilon, s > 0$ and $p \leq p(w,s) \equiv c\delta^{s/2w}/(w^3\log(1/\varepsilon))^2$. For any width w DNF $f : \{0,1\}^n \to \{0,1\}$ and $\rho \leftarrow \mathcal{D} \in \mathcal{D}_p(w)$, with probability at least $1 - \delta^{s/4} - \varepsilon$ there exist width $w/2$ DNFs $f^\ell, f^u : \{0,1\}^{L_\rho} \to \{0,1\}$ such that $f^\ell \leq f_\rho \leq f^u$ and $Bias(f^u) - Bias(f^\ell) < \varepsilon$.*

Proof of Theorem II.3: As described in the introduction, the proof will be by alternately applying Theorem I.1 and Claim VI.1.

Let t be such that $w/2^t = s$ and for $r = 1,\ldots,t$, let $p_r = p(w/2^r, s)$ as defined in the above claim. For $i \in [t]$, let L_i be chosen independently from a distribution in $\mathcal{D}_{p_i}(w/2^i)$. Let $L = \cap_{i=1}^t L_i$ and for $x \in_u \{0,1\}^n$, let $\rho = \rho_{L,x}$. Then, ρ is a random restriction with indicator variables $1\{i \in L\}$ having bias

$$q = p_1 p_2 \cdots p_t > \frac{c^{\log w} \cdot \delta^{\sum_{r=1}^t s2^r/2w}}{(w^3\log(1/\varepsilon))^{\log w}} > \frac{\delta}{(w\log(1/\varepsilon))^{C\log w}},$$

for C a sufficiently large constant.

Define the composition of two restrictions $\rho' \in \{*,0,1\}^L$ and $\rho'' \in \{*,0,1\}^{L(\rho')}$ in the natural way by $(\rho' \circ \rho'')_i = \rho''_i$ if $i \in L(\rho')$ and $(\rho' \circ \rho'')_i = \rho'_i$ otherwise. Then, by definition, we can view ρ as a composition of independently chosen random restrictions $\rho_t \circ \rho_{t-1} \circ \cdots \circ \rho_1$, where $\rho_j \equiv \rho_{L_j, x^j}$ (with $x^j \in_u \{0,1\}^n$). Further, for any function g, $g_\rho \equiv (((g_{\rho_1})_{\rho_2})\ldots)_{\rho_t}$.

Therefore, by iteratively applying the Claim VI.1 t times with the random restrictions $\rho_1,\ldots,\rho_t$ and a union bound, we get that with probability at least $1 - t(\delta^{s/4} + \varepsilon)$, there exists a lower approximating width at most $w/2^{t+1}$ DNF $f^\ell : \{0,1\}^L \to \{0,1\}$ such that $f^\ell \leq f_\rho$ and $Bias(f_\rho) - Bias(f^\ell) < t\varepsilon$. Similarly, by iteratively applying the claim to the upper approximators given by the claim, we get that with probability at least $1 - 2t(\delta^{s/4} + \varepsilon)$, f_ρ is $(t\varepsilon)$-strongly approximable by width at most s DNFs.

Finally, observe that the number of bits needed to sample L is

$$\sum_{r=1}^t O\left(\frac{w}{2^r} \cdot \log(1/p(w/2^r, s)) + \log n\right)$$
$$= O\left((\log n)(\log w)\right) +$$
$$\sum_{r=1}^t \frac{w}{2^r}\left(\frac{s2^r}{2w} O\left(\log(1/\delta)\right) + O\left(\log(w\log(1/\varepsilon))\right)\right)$$
$$= O\left((\log w)\cdot(\log n + s\log(1/\delta)) + w\log\left(w\log(1/\varepsilon)\right)\right).$$

The theorem now follows from applying the above argument to $\delta' = \delta/2t$, $\varepsilon' = \varepsilon/t$ and noting that this only changes the constant terms in the final bounds. ■

Proof of Claim VI.1: Let f^ℓ, f^u be width w DNFs with at most $h(w) = w^{3w}(C\log(1/\varepsilon))^w$ terms that $\varepsilon^2/2$-strongly approximate f as guaranteed by Theorem I.1

for C a large constant. Consider a random restriction ρ sampled from a distribution in $\mathcal{D}_p(w/2)$. Then, the probability that a fixed term of f^ℓ has more than $w/2$ live variables under ρ is at most $2^w \cdot p^{w/2}$. Therefore, by a union bound, the probability that f^ℓ_ρ has width more than $w/2$ is at most $h(w)2^w p^{w/2} < \delta^{s/4}/2$ for a sufficiently small constant c. Similarly, the probability that f^u_ρ has width more than $w/2$ is at most $\delta^{s/4}/2$.

Note that as $f^\ell \leq f \leq f^u$, $f^\ell_\rho \leq f_\rho \leq f^u_\rho$. We now need to show that f^ℓ_ρ, f^u_ρ are close to f_ρ with high probability. Let $\rho \equiv \rho_{L,x}$ and consider a fixing of the set of live variables L. Then as f^ℓ, f^u strongly approximate f,

$$Bias(f) = \mathop{\mathbb{E}}_{x \leftarrow \{0,1\}^{[n]\setminus L}}[Bias(f_\rho)] < Bias(f^\ell) + \varepsilon = \mathop{\mathbb{E}}_{x \leftarrow \{0,1\}^{[n]\setminus L}}[Bias(f^\ell_\rho)] + \varepsilon^2/2.$$

Therefore, $\mathbb{E}_{x \leftarrow \{0,1\}^{[n]\setminus L}}[Bias(f_\rho) - Bias(f^\ell_\rho)] < \varepsilon^2/2$. Thus, by Markov's inequality,

$$\Pr_{x \leftarrow \{0,1\}^{[n]\setminus L}}[Bias(f_\rho) - Bias(f^\ell_\rho) > \varepsilon] < \varepsilon/2.$$

Using a similar argument to f^u, and a union bound, we get that f_ρ is not ε-strongly approximable by (f^ℓ_ρ, f^u_ρ) with probability at most $\delta^{s/4} + \varepsilon$. The claim now follows. ∎

VII. Open Problems

The obvious open problem is to obtain a deterministic polynomial time algorithm as opposed to a $n^{\tilde{O}((\log\log n)^2)}$ algorithm for approximately counting the solutions to DNFs. Other avenues are to ask for better dependence on the error rate ε. This is particularly interesting for $\varepsilon = 1/\mathrm{poly}(n)$ as from the work of Karp and Luby [7] it follows that a deterministic approximate counting algorithm for DNFs with additive error $1/\mathrm{poly}(n)$ implies a deterministic approximate counting algorithm with small *relative* error (i.e., given a DNF f, compute p such that $(1-\varepsilon)p < Bias(f) < (1+\varepsilon)p$).

It is not clear what the right bound on W should be. The proof of Theorem I.1 is based on a variant of the sunflower lemma, the widely believed sunflower conjecture (cf. [21] for instance) and Friedgut's theorem suggests that the right bound might be $2^{O_\varepsilon(w)}$. An example showing that the number of terms needed for a good approximation is at least 4^w was communicated to us by Servedio [22]. We believe that getting tight bounds on W is an interesting question in its own right.

Acknowledgements

We thank Adam Klivans, Rocco Servedio, Avi Wigderson and David Zuckerman for valuable discussions. We thank Rocco for drawing our attention to Friedgut's theorem in this context.

References

[1] L. M. J. Bazzi, "Polylogarithmic independence can fool DNF formulas," *SIAM J. Comput.*, vol. 38, no. 6, pp. 2220–2272, 2009.

[2] B. Rossman, "The monotone complexity of k-clique on random graphs," in *FOCS*, 2010, pp. 193–201.

[3] L. Trevisan, "A note on approximate counting for k-DNF," in *APPROX-RANDOM*, 2004, pp. 417–426.

[4] M. Ajtai and A. Wigderson, "Deterministic simulation of probabilistic constant depth circuits (preliminary version)," in *FOCS*, 1985, pp. 11–19.

[5] E. Friedgut, "Boolean functions with low average sensitivity depend on few coordinates," *Combinatorica*, vol. 18, no. 1, pp. 27–35, 1998.

[6] K. Amano, "Tight bounds on the average sensitivity of k-cnf," *Theory of Computing*, vol. 7, no. 1, pp. 45–48, 2011.

[7] R. M. Karp and M. Luby, "Monte-carlo algorithms for enumeration and reliability problems," in *FOCS*, 1983, pp. 56–64.

[8] N. Nisan and A. Wigderson, "Hardness vs randomness," *J. Comput. Syst. Sci.*, vol. 49, no. 2, pp. 149–167, 1994.

[9] N. Nisan, "Pseudorandom bits for constant depth circuits," *Combinatorica*, vol. 11, no. 1, pp. 63–70, 1991.

[10] M. Luby and B. Velickovic, "On deterministic approximation of DNF," in *STOC*, 1991, pp. 430–438.

[11] ——, "On deterministic approximation of DNF," *Algorithmica*, vol. 16, no. 4/5, pp. 415–433, 1996.

[12] M. Luby, B. Velickovic, and A. Wigderson, "Deterministic approximate counting of depth-2 circuits," in *ISTCS*, 1993, pp. 18–24.

[13] A. A. Razborov, "A simple proof of bazzi's theorem," *TOCT*, vol. 1, no. 1, 2009.

[14] A. De, O. Etesami, L. Trevisan, and M. Tulsiani, "Improved pseudorandom generators for depth 2 circuits," in *APPROX-RANDOM*, 2010, pp. 504–517.

[15] L. Valiant, "The complexity of computing the permanent," *Theoretical Computer Science*, vol. 8, no. 2, pp. 189 – 201, 1979.

[16] N. Linial and N. Nisan, "Approximate inclusion-exclusion," *Combinatorica*, vol. 10, pp. 349–365, 1990.

[17] M. Braverman, "Polylogarithmic independence fools ac^0 circuits," *J. ACM*, vol. 57, no. 5, 2010.

[18] J. Håstad, "Almost optimal lower bounds for small depth circuits," in *STOC*, 1986, pp. 6–20.

[19] M. Agrawal, E. Allender, R. Impagliazzo, T. Pitassi, and S. Rudich, "Reducing the complexity of reductions," *Computational Complexity*, vol. 10, no. 2, pp. 117–138, 2001.

[20] J. Naor and M. Naor, "Small-bias probability spaces: Efficient constructions and applications," *SIAM J. Comput.*, vol. 22, no. 4, pp. 838–856, 1993.

[21] N. Alon, A. Shpilka, and C. Umans, "On sunflowers and matrix multiplication," in *CCC*, 2012, to appear.

[22] R. Servedio, 2011, personal communication.

APPENDIX

We first that Lemma III.1 is equivalent to Lemma A.2 below from [2].

Definition A.1 ([2]). *Let $\mathcal{F}$ be a family of sets over a universe U and let $Y = \cap_{T\in\mathcal{F}} S$. Call $\mathcal{F}$ a γ-sunflower if for a random set $W \subseteq U$, with each element of U present in W independently with probability $1/2$, $\mathsf{Pr}[\forall T \in \mathcal{F}, T \cap (W \setminus Y) \neq \emptyset] < \gamma$.*

Lemma A.2 ([2]). *Let $\mathcal{F}$ be a family of sets over a universe U each of size at most w. If $|\mathcal{F}| > w! \cdot (2.47\log(1/\gamma))^w$, then $\mathcal{F}$ contains a γ-sunflower.*

Proof of Lemma III.1: As f is unate, without loss of generality suppose that f is monotone. Let $U = [n]$ and $\mathcal{F} = \{T_i : 1 \le i \le m\}$. By the above lemma, there exists a γ-sunflower $\mathcal{F}' = \{T_{i_1}, \ldots, T_{i_s}\}$ for $\gamma = \exp(-(m/w!)^{1/w}/2.47)$. We claim that the lemma holds for the terms in $\mathcal{F}'$ and $Y = \cap_{j=1}^{s} T_{i_j}$. Let $x \in_u \{0,1\}^n$ and let $W = \{i : x_i = 0\}$. Then, each element of U is present in W independently with probability $1/2$. Therefore, as $\mathcal{F}'$ is a γ-sunflower

$$\mathsf{Pr}[x \text{ satisfies } (T_{i_1} \setminus Y) \vee (T_{i_2} \setminus Y) \cdots \vee (T_{i_s} \setminus Y)] = \mathsf{Pr}[\,\exists T \in \mathcal{F}',\ (T \setminus Y) \cap W = \emptyset\,] \ge 1 - \gamma.$$

■

Proof of Theorem I.1: We now sketch the missing calculations from the proof of the theorem:

$$\begin{aligned}\sum_{j=W+1}^{m} j \cdot \delta_w(j) &= \sum_{j=W+1}^{m} j \cdot \exp\left(\frac{-(j/2^w w!)^{1/w}}{c}\right)\\ &\le \sum_{j=W+1}^{m} j \cdot \exp\left(\frac{-j^{1/w}}{2cw}\right)\\ &\le \sum_{r=1}^{\infty} (e^r W)^2 \cdot \exp\left(\frac{-(e^{r-1}\cdot W)^{1/w}}{2cw}\right)\\ &\le W^2 \sum_{r=1}^{\infty} \exp\left(\frac{-(e^{r-1}\cdot W)^{1/w}}{2cw} + 2r\right).\end{aligned}$$

We next check that the sum decreases exponentially:

$$\frac{(e^{r-1}\cdot W)^{1/w}}{2cw} - 2r \ge \frac{W^{1/w}}{2cw} + r,$$

which happens if

$$\frac{W^{1/w}}{2cw} \cdot \left(e^{(r-1)/w} - 1\right) \ge 3r.$$

The last inequality is easy to check for our setting of W Therefore, we get (recall that $c = 2.47$)

$$\begin{aligned}\sum_{j=W+1}^{m} j \cdot \delta_w(j) &\le W^2 \cdot \exp\left(-W^{1/w}/2cw\right)\\ &\le w^{6w}(50\log(1/\varepsilon))^{2w} \cdot \exp\left(-10\cdot w^2 \cdot \log(1/\varepsilon)\right)\\ &\le \exp\left(-10\cdot w^2\cdot\log(1/\varepsilon) + (6w\log w) + (2w)\cdot(\log(50\log(1/\varepsilon)))\right)\\ &\le \exp\left(-2w^2\log(1/\varepsilon)\right),\end{aligned}$$

where the last inequality follows from: $3w^2 \ge 6w(\log w)$; $5\log(1/\varepsilon) \ge 2\log(50\log(1/\varepsilon))$ if $\varepsilon < 1/6$

Equations III.2, III.3 follow easily from the above. ■