THEORY OF COMPUTING, Volume 9 (7), 2013, pp. 283–293
www.theoryofcomputing.org

Pseudorandomness for Width-2 Branching Programs

Andrej Bogdanov* Zeev Dvir† Elad Verbin‡ Amir Yehudayoff§

Received September 4, 2009; Revised October 10, 2012; Published February 27, 2013

Abstract: We show that pseudorandom generators that fool degree-k polynomials over $\mathbb{F}_2$ also fool branching programs of width-2 and polynomial length that read k bits of input at a time. This model generalizes polynomials of degree k over $\mathbb{F}_2$ and includes some other interesting classes of functions, for instance, k-DNFs.

The proof essentially follows by a new decomposition theorem for width-2 branching programs. The theorem states that if f can be computed by a width-2 branching program that reads k bits of input at a time, then f can be (roughly) written as a sum $f = \sum_i \alpha_i f_i$ where each f_i is a degree-k polynomial and $\sum_i |\alpha_i|$ is small.

Bogdanov and Viola (FOCS 2007) constructed a pseudorandom generator that fools degree-k polynomials over $\mathbb{F}_2$ for arbitrary k. Their construction consists of summing k independent copies of a generator that ε-fools linear functions. Our second result investigates the limits of such constructions: We show that, in general, such a construction is not pseudorandom against bounded fan-in circuits of depth $O((\log(k \log 1/\varepsilon))^2)$.

ACM Classification: G.3

AMS Classification: 68W20

Key words and phrases: pseudorandomness, branching programs, harmonic analysis

*Work done while at Tsinghua University. Supported in part by the National Basic Research Program of China Grants 2007CB807900, 2007CB807901 and Hong Kong RGC GRF CUHK410111 and CUHK410112.

†Research partially supported by NSF grant CCF-0832797.

‡Supported in part by the National Natural Science Foundation of China Grant 60553001, and the National Basic Research Program of China Grants 2007CB807900, 2007CB807901.

§Horev fellow—supported by the Taub Foundation. Research supported by ISF and BSF.

DOI: 10.4086/toc.2013.v009a007

1 Introduction

This work studies the power and limitation of certain pseudorandom distributions and pseudorandom generators. Let us begin with a definition. A distribution D on $\{0,1\}^n$ is *ε-pseudorandom* against a class C of functions from $\{0,1\}^n$ to $\{0,1\}$ if for every $f \in C$,

$$\Big|\Pr_{x\sim D}[f(x)=1] - \Pr_{x\sim\{0,1\}^n}[f(x)=1]\Big| \le \varepsilon\,,$$

where $x \sim \{0,1\}^n$ means that x is uniformly distributed in $\{0,1\}^n$. A function $G : \{0,1\}^m \to \{0,1\}^n$ is *an ε-pseudorandom generator* (PRG) against C if the distribution $G(s)$, $s \sim \{0,1\}^m$, is ε-pseudorandom against C. We call m the *seed length* of the generator.

Bogdanov and Viola [4] suggested the following construction of a PRG against degree-k polynomials over a finite field:

$$G'(s_1,\ldots,s_k) = G(s_1) + \cdots + G(s_k)\,, \tag{1.1}$$

where G is a PRG against linear functions [11]. Building on their work and subsequent work by Lovett [9], Viola [16] proved that this generator is indeed pseudorandom against low-degree polynomials: If G is ε-pseudorandom against linear functions, then G' is $O(\varepsilon^{1/2^{k-1}})$-pseudorandom against degree-k polynomials. By Viola's analysis, this pseudorandom generator has seed length that is optimal up to a constant factor (as long as the degree of the polynomial and ε are constant). However, the seed length deteriorates exponentially with the degree, and the result becomes trivial when the degree exceeds $\log n$.

Is the Bogdanov-Viola construction also pseudorandom for polynomials of, say, polylogarithmic degree? A positive answer would give a new pseudorandom generator against polynomial-size constant-depth circuits with modular gates, since such circuits can be approximated by polynomials of poly-logarithmic degree in a very strong sense [12, 14]. In other words, it would give an example where a pseudorandom generator that was designed for one class of functions (polynomials) would automatically yield a derandomization of a different class (small-depth circuits). Conversely, if we believe that the Bogdanov-Viola generator is *not* pseudorandom against polynomials of polylogarithmic degree, we could try proving this by giving a constant-depth circuit against which the generator is not pseudorandom.

More generally, given a probability distribution with some property (k-wise independence, small bias against linear functions, a convolution of several such distributions), is it pseudorandom against some class of functions (small circuits, space bounded computations)? Such questions have been considered before. For example, Linial and Nisan [8] showed that DNFs on n variables are fooled by $O(\sqrt{n}\log n)$-wise independent distributions and conjectured that polylogarithmic-wise independence suffices. A beautiful sequence of papers improved our knowledge: Bazzi proved the Linial-Nisan conjecture for depth-2 circuits [2], Razborov simplified Bazzi's proof [13], and Braverman proved the conjecture for constant-depth circuits [6].

This line of study could reveal insights about the power and limitations of existing constructions of pseudorandom generators. In this paper we prove two results, one positive and one negative.

1.1 A positive result

Here we are interested in the class (k,t,n)-2BP of *width-2 branching programs* of length t that read k bits of input at a time and compute a function from $\{0,1\}^n$ to $\{0,1\}$. This device can be described by a layered directed acyclic graph, where there are t layers and each layer contains two nodes, which we label by 0 and 1. Each layer $j < t$ is associated with an arbitrary k-bit substring $x|_j$ of the input x. Each node in layer j has 2^k outgoing edges to layer $j+1$ that are labelled by all possible values in $\{0,1\}^k$. On input x, the computation starts with the first node in the first layer, then follows the edge labelled by $x|_1$ onto the second layer, and so on until a node in the last layer is reached. The identity of this last node is the outcome of the computation.

This type of branching program can represent a degree-k polynomial, a "space-bounded" computation with one bit of memory, as well as a k-DNF formula.[1] We prove the following.

Theorem 1.1. *Let G be an ε-PRG against degree-k polynomials in n variables over $\mathbb{F}_2$. Then G is an ε'-PRG against the class of functions computed by a (k,t,n)-2BP, with $\varepsilon' = t \cdot \varepsilon$.*

Most of the literature on PRGs for branching programs deals with read-once programs, due to their connection to space-bounded computation. We consider general branching programs that are not necessarily read-once. General programs are interesting also due to Barrington's theorem [1] which states that any circuit of depth d can be simulated by a (non-read-once) branching program of width 5 and size 4^d.

The fooling parameter ε' for branching programs in the theorem is t times the fooling parameter ε for degree-k polynomials. A typical value for t is polynomial in the number of variables. This means that in this case (according to the theorem above) only an inverse-polynomial error for polynomials implies a non-trivial statement for branching programs.

1.2 A negative result

Our second result shows limitations of the Bogdanov-Viola construction.

Theorem 1.2. *Let k be an integer and $\varepsilon > 0$. Let $m = k\log(1/\varepsilon)+1$. For every $n \geq 2m^2$, there exists a distribution D such that D is ε-pseudorandom against linear functions over $\{0,1\}^n$, but the sum of k independent copies of D is not $1/3$-pseudorandom against bounded fan-in circuits (with* and*, * or*, and* not *gates) of depth $O(\log^2 m)$ and size polynomial in m.*

It is known [11] that the seed length of an ε-biased generator against linear functions must be at least $\Omega(\log n + \log(1/\varepsilon))$. Therefore, if we want the generator to be efficient, we are restricted to using $\varepsilon = 1/\mathrm{poly}(n)$. For this setting of parameters, Theorem 1.2 tells us that the Bogdanov-Viola generator does not fool bounded fan-in circuits of depth $O((\log(k\log n))^2)$. Barrington's theorem implies, therefore, that D^k is not pseudorandom against width-5 size-$2^{O((\log(k\log 1/\varepsilon))^2)}$ branching programs. Concluding, the Bogdanov-Viola generator fools branching programs of width 2, but not of width 5. We do not know what happens for width 3 or 4.

A correlation bound of Viola and Wigderson [17] shows that, in general, a distribution that is pseudorandom against degree-k polynomials need not be pseudorandom against the "$\equiv 0 \mod 3$" function, which is computable by a width-3 branching program. However, their (counter-)example is not a convolution of independent distributions with small bias against linear functions (i. e., it does not have the form of the Bogdanov-Viola generator).

[1] In the special case of k-DNFs, Trevisan [15] has a better result. He showed that for every constant k there is an $\varepsilon = 1/\mathrm{poly}(n)$ such that ε-biased generators against linear functions fool k-DNF over n variables.

1.3 Proof overview for Theorem 1.1

It has been known for some time that *read-once* width-2 branching programs that read one bit at a time can be fooled by linear generators.[2] One way to argue this is to think of the computation of the branching program B as a boolean function over $\mathbb{F}_2^n$ and show inductively over the layers of B that the sum of the absolute values of the Fourier coefficients of B is bounded from above by t. It is easy to see that linear generators of bias ε are εL-pseudorandom against any boolean function whose sum of absolute values of Fourier coefficients is at most L, and the correctness follows from there.

For branching programs that read more than one bit at a time, this argument cannot work, as there exist width-2 branching programs that read 2 bits at a time and that are not fooled by some small-bias linear generator. One such branching program computes the inner product function (for even n)

$$IP(x_1,\ldots,x_n) = x_1x_2 + \cdots + x_{2i-1}x_{2i} + \cdots + x_{n-1}x_n \pmod 2 .$$

Nevertheless, we argue along the same lines. Instead of using the Fourier transform of the branching program, we resort to "higher-order" representations of functions using low-degree polynomials. We show that every branching program B of length t and width 2 that reads k bits at a time admits a "representation of length t" in terms of degree-k polynomials. By "representation of length t" we mean that B can be written as a sum *over the reals* of the form

$$(-1)^{B(x)} = \sum_{p:\mathbb{F}_2^n \to \mathbb{F}_2} \alpha_p \cdot (-1)^{p(x)}$$

where p ranges over all degree-k polynomials over $\mathbb{F}_2$, and α_p are real coefficients such that $\sum_p |\alpha_p| \le t$. Unlike the Fourier transform, for degree 2 and larger this representation is not unique. Once this representation has been obtained, we argue that a pseudorandom generator for degree-k polynomials is also pseudorandom for B by linearity of expectation.

While our proof is not technically difficult, we find the application of "higher-order" Fourier type analysis conceptually interesting and potentially relevant to other computer science applications.

2 Fooling width-2 branching programs

2.1 Width-2 branching programs as sums of polynomials

The following theorem is the basis for the proof of Theorem 1.1. It shows that width-2 branching programs have a "short representation by polynomials of low degree." For a function $f : \{0,1\}^n \to \{0,1\}$, we use the notation $\tilde{f} = (-1)^f$, a map from $\{0,1\}^n$ to $\{1,-1\}$. Define $\deg(f)$ to be the degree of f when viewed as a multilinear polynomial in $\mathbb{F}_2[x_1,\ldots,x_n]$.

Theorem 2.1. *Let $f : \{0,1\}^n \to \{0,1\}$ be computed by a (k,t,n)-2BP. Then there exist $\alpha_1,\ldots,\alpha_s \in \mathbb{R}$ and $g_1,\ldots,g_s : \{0,1\}^n \to \{0,1\}$ such that*

1. $\tilde{f}(x) = \sum_{i=1}^{s} \alpha_i \cdot \tilde{g}_i(x)$ *for all $x \in \{0,1\}^n$ (where the sum is over the reals),*

[2] We are not aware of a published proof but have heard the result credited to Saks and Zuckerman.

2. *for all* $i \in [s]$, $\deg(g_i) \le k$, *and*

3. $\sum_{i=1}^{s} |\alpha_i| \le t$.

We defer the proof of Theorem 2.1 to Section 2.2 and proceed by showing how it implies our main result.

Proof of Theorem 1.1. Let $G : \{0,1\}^m \to \{0,1\}^n$ be an ε-pseudorandom generator against degree-k polynomials in n variables over $\mathbb{F}_2$. Let $f : \{0,1\}^n \to \{0,1\}$ be computed by a (k,t,n)-2BP. By Theorem 2.1, there exist $\alpha_1,\ldots,\alpha_s \in \mathbb{R}$ and $g_1,\ldots,g_s : \{0,1\}^n \to \{0,1\}$ such that

1. $\tilde{f}(x) = \sum_{i=1}^{s} \alpha_i \cdot \tilde{g}_i(x)$ for all $x \in \{0,1\}^n$;

2. for all $i \in [s]$, $\deg(g_i) \le k$;

3. $\sum_{i=1}^{s} |\alpha_i| \le t$.

For the rest of the proof $x \sim \{0,1\}^n$ and $s \sim \{0,1\}^m$ denote two independent random variables chosen uniformly from their respective domains. First, note that for every function $f : \{0,1\}^n \to \{0,1\}$,

$$2 \cdot \big|\Pr[f(G(s))=1] - \Pr[f(x)=1]\big| = \big|\mathbf{E}[\tilde{f}(G(s)) - \tilde{f}(x)]\big| .$$

Thus, using the properties above, and using linearity of expectation,

$$\begin{aligned}
2 \cdot \big|\Pr[f(G(s))=1] - \Pr[f(x)=1]\big| &= \big|\mathbf{E}\big[\tilde{f}(G(s)) - \tilde{f}(x)\big]\big| \\
&= \Big|\sum_{i=1}^{s} \alpha_i \cdot \mathbf{E}\big[\tilde{g}_i(G(s)) - \tilde{g}_i(x)\big]\Big| \\
&\le \sum_{i=1}^{s} |\alpha_i| \cdot \big|\mathbf{E}\big[\tilde{g}_i(G(s)) - \tilde{g}_i(x)\big]\big| \\
&= \sum_{i=1}^{s} |\alpha_i| \cdot 2 \cdot \big|\Pr[g_i(G(s))=1] - \Pr[g_i(x)=1]\big| \\
&\le 2 \cdot t \cdot \varepsilon ,
\end{aligned}$$

where the last inequality holds since G is an ε-pseudorandom generator against degree-k polynomials. □

2.2 Proof of Theorem 2.1

Let f be a boolean function computed by a branching program B of width 2 and length t that reads k bits of input at a time. We prove the theorem by induction on t.

Induction base: For the case $t \le 2$, the theorem holds since $f(x)$ is a boolean function in at most k variables and so $\deg(f) \le k$.

Induction step: Assume that the theorem holds for every function computed by a $(k,t-1,n)$-2BP. By definition, there exists $P : \{0,1\}^{k+1} \to \{0,1\}$ such that

$$f(x) = P\left(f_{t-1}(x), x|_{t-1}\right) ,$$

where f_{t-1} is the function computed at layer $(t-1)$ of B, and $x|_{t-1}$ is the k-bit substring of the input x associated with layer $(t-1)$.

Let p_0 and p_1 be two maps from $\{0,1\}^k$ to $\{0,1\}$ defined as

$$p_0(y) = P(0,y) \quad \text{and} \quad p_1(y) = P(1,y)\,.$$

Since both of p_0 and p_1 depend on at most k variables, we know $\deg(p_0) \le k$ and $\deg(p_1) \le k$. In addition, for every $z \in \{0,1\}$ and $y \in \{0,1\}^k$,

$$\tilde{P}(z,y) = \frac{1}{2}(\tilde{p}_0(y) - \tilde{p}_1(y)) \cdot (-1)^z + \frac{1}{2}(\tilde{p}_0(y) + \tilde{p}_1(y))\,.$$

We can now use the induction hypothesis. By the choice of P, for every $x \in \{0,1\}^n$,

$$\tilde{f}(x) = \tilde{P}(f_{t-1}(x), x|_{t-1}) = \frac{1}{2}\left(\tilde{p}_0(x|_{t-1}) - \tilde{p}_1(x|_{t-1})\right) \cdot \tilde{f}_{t-1}(x) + \frac{1}{2}\left(\tilde{p}_0(x|_{t-1}) + \tilde{p}_1(x|_{t-1})\right)\,.$$

By induction, there exist $\alpha_1, \ldots, \alpha_s \in \mathbb{R}$ and $g_1, \ldots, g_s : \{0,1\}^n \to \{0,1\}$ that correspond to $\tilde{f}_{t-1}(x)$ and satisfying the three stated properties. Thus, for all $x \in \{0,1\}^n$

$$\tilde{f}(x) \quad = \quad \frac{1}{2}\left(\tilde{p}_0(x|_{t-1}) - \tilde{p}_1(x|_{t-1})\right) \cdot \sum_{i=1}^{s} \alpha_i \cdot \tilde{g}_i(x) + \frac{1}{2}\left(\tilde{p}_0(x|_{t-1}) + \tilde{p}_1(x|_{t-1})\right).$$

We complete the proof by renaming the polynomials and the coefficients in the above sum. For $j \in \{1, \ldots, s\}$, set

$$\beta_j = \frac{\alpha_j}{2} \quad \text{and} \quad h_j(x) = p_0(x|_{t-1}) \oplus g_j(x)$$

and for $j \in \{s+1, \ldots, 2s\}$, set

$$\beta_j = -\frac{\alpha_{j-s}}{2} \quad \text{and} \quad h_j(x) = p_1(x|_{t-1}) \oplus g_j(x)$$

(where $\oplus$ denotes summation in $\mathbb{F}_2$). Set $\beta_{2s+1} = \beta_{2s+2} = 1/2$, set $h_{2s+1}(x) = p_0(x|_{t-1})$, and set $h_{2s+2}(x) = p_1(x|_{t-1})$. Finally, set $s' = 2s+2$. Thus,

$$\tilde{f}(x) = \sum_{j=1}^{s'} \beta_j \cdot \tilde{h}_j(x)$$

for all $x \in \{0,1\}^n$. In addition, every h_j is of degree at most k (since addition in $\mathbb{F}_2$ does not increase the degree), and

$$\sum_{j=1}^{s'} |\beta_j| \le 1 + 2 \cdot \sum_{i=1}^{s} \frac{|\alpha_i|}{2} \le 1 + (t-1) = t\,.$$

□

3 Limitations of the Bogdanov-Viola construction

In this section we show that a sum of several copies of pseudorandom generators for linear functions fails to fool small-depth, bounded fan-in circuits (Theorem 1.2).

3.1 Proof of Theorem 1.2

Set $m = k\log(1/\varepsilon)+1$ and partition the input $x \in \mathbb{F}_2^n$ into n/m consecutive blocks $x|_1, \dots, x|_{n/m} \in \mathbb{F}_2^m$. Consider the following distribution D.

1. Choose a random linear subspace S of $\mathbb{F}_2^m$ of dimension $(m-1)/k$.
2. For $1 \le i \le n/m$, choose each block $x|_i$ independently and uniformly from S.

To prove Theorem 1.2, we show the following two claims.

Claim 3.1. *The distribution D is ε-pseudorandom against linear functions.*

Claim 3.2. *The sum D^k of k independent samples from D is not $1/3$-pseudorandom against bounded fanin circuits of depth $O((\log m)^2)$ and size polynomial in m.*

The theorem follows from these two claims.

Proof of Claim 3.1. Let $a(x) = \langle a, x\rangle$ be an arbitrary nonzero linear function over $\mathbb{F}_2^n$. We split a as a sum of linear functions a_i over the blocks of x as

$$a(x) = \sum_{i=1}^{n/m} a_i(x|_i)\,.$$

Without loss of generality, let us assume a_1 is nonzero. Conditioned on the choice of S, the values of the functions $a_i(x|_i)$ are independent:

$$\mathop{\mathbf{E}}_{x\sim D}[(-1)^{a(x)}] = \mathop{\mathbf{E}}_{S}\Big[\prod_{i=1}^{n/m} \mathop{\mathbf{E}}_{x|_i\sim S}[(-1)^{a_i(x|_i)}]\Big]\,.$$

Now for any fixed choice of S, the value $\mathbf{E}_{x|_i\sim S}[(-1)^{a_i(x|_i)}]$ is 1 if $a_i \in S^\perp$ and 0 otherwise. Here

$$S^\perp = \{y : \langle y, x\rangle = 0 \text{ for all } x \in S\}\,.$$

Therefore

$$|\mathop{\mathbf{E}}_{x\sim D}[(-1)^{a(x)}]| = \mathbf{Pr}[\text{for all } i,\ a_i \in S^\perp] \le \mathbf{Pr}[a_1 \in S^\perp] = 2^{-(m-1)/k} = \varepsilon$$

and so $|\mathbf{E}_{x\sim D}[a(x)] - 1/2| \le \varepsilon/2 < \varepsilon$. □

Proof of Claim 3.2. Let $X_1, \dots, X_k$ be independent samples from the distribution D and $X = X_1 + \cdots + X_k$. Let S_i denote the subspace of $\mathbb{F}_2^m$ associated to the sample X_i. Since each block of X_i belongs to the subspace S_i, each block of X will belong to the sum of subspaces $S = S_1 + \cdots + S_k$. The subspace S has dimension at most $m-1$.

This suggests the following test for X: Arrange the first $2m$ blocks of X as rows in an $m \times 2m$ matrix M and compute the rank of M over $\mathbb{F}_2$. (By our choice of parameters, $2m^2 \le n$ so this is always possible.) If the matrix has full rank, output 1, otherwise output 0. If X is chosen from D^k, then all columns of M are chosen from the same subspace of dimension $m-1$ so M will never have full rank. If X is chosen from the uniform distribution, then M is a random $m \times 2m$ matrix and, by a standard argument, the probability it doesn't have full rank is at most $2^{-m} < 1/3$.

It remains to observe that the above test, which is essentially a rank computation, can be implemented by a circuit of depth $O((\log m)^2)$ and size polynomial in m, using a theorem of Mulmuley [10] (see also earlier work by Csanky [7], Berkowitz [3], and Borodin et al. [5]). $\square$

4 Open Problems

First, the question of building a generator with seed length $o(\log^2 n)$ that fools width-3 branching programs is wide open. It is open even for *read-once* width-3 branching programs.

Second, as discussed in Section 1, it would be interesting to find a generator that fools read-once degree-k polynomials, that has shorter seed-length than Lovett's generator. (Recall that Lovett's generator also fools non-read-once polynomials).

5 Acknowledgments

We thank Anup Rao for helpful conversations on this problem. This work was done while the authors took part in the "China Theory Week" workshop at Tsinghua University. We would like to thank the organizers of the workshop and in particular Andy Yao for their hospitality.

References

[1] DAVID A. MIX BARRINGTON: Bounded-width polynomial-size branching programs recognize exactly those languages in NC^1. *J. Comput. System Sci.*, 38(1):150–164, 1989. Preliminary version in STOC'86. [doi:10.1016/0022-0000(89)90037-8] 285

[2] LOUAY M. J. BAZZI: Polylogarithmic independence can fool DNF formulas. *SIAM J. Comput.*, 38(6):2220–2272, 2009. Preliminary version in FOCS'07. [doi:10.1137/070691954] 284

[3] STUART J. BERKOWITZ: On computing the determinant in small parallel time using a small number of processors. *Inform. Process. Lett.*, 18(3):147–150, 1984. [doi:10.1016/0020-0190(84)90018-8] 290

[4] ANDREJ BOGDANOV AND EMANUELE VIOLA: Pseudorandom bits for polynomials. *SIAM J. Comput.*, 39(6):2464–2486, 2010. Preliminary version in FOCS'07. [doi:10.1137/070712109] 284

[5] ALLAN BORODIN, JOACHIM VON ZUR GATHEN, AND JOHN E. HOPCROFT: Fast parallel matrix and GCD computations. *Inf. Control*, 52(3):241–256, 1982. Preliminary version in FOCS'82. [doi:10.1016/S0019-9958(82)90766-5] 290

[6] MARK BRAVERMAN: Polylogarithmic independence fools AC^0 circuits. *J. ACM*, 57(5):28, 2010. Preliminary version in CCC'09, exposition in Comm. ACM. [doi:10.1145/1754399.1754401] 284

[7] LASZLO CSANKY: Fast parallel matrix inversion algorithms. *SIAM J. Comput.*, 5(4):618–623, 1976. Preliminary version in FOCS'75. [doi:10.1137/0205040] 290

[8] NATHAN LINIAL AND NOAM NISAN: Approximate inclusion-exclusion. *Combinatorica*, 10(4):349–365, 1990. Preliminary version in STOC'90. [doi:10.1007/BF02128670] 284

[9] SHACHAR LOVETT: Unconditional pseudorandom generators for low degree polynomials. *Theory of Computing*, 5(3):69–82, 2009. Preliminary version in STOC'08. [doi:10.4086/toc.2009.v005a003] 284

[10] KETAN MULMULEY: A fast parallel algorithm to compute the rank of a matrix over an arbitrary field. *Combinatorica*, 7(1):101–104, 1987. Preliminary version in STOC'86. [doi:10.1007/BF02579205] 290

[11] JOSEPH NAOR AND MONI NAOR: Small-bias probability spaces: Efficient constructions and applications. *SIAM J. Comput.*, 22(4):838–856, 1993. Preliminary version in STOC'90. [doi:10.1137/0222053] 284, 285

[12] ALEXANDER A. RAZBOROV: Lower bounds on the size of bounded depth circuits over a complete basis with logical addition. *Matematicheskie Zametki*, 41:598–607, 1987. English translation: Math. Notes of the Acad. Sci. USSR, Vol. 41 (4), 1987, pp. 333-338. 284

[13] ALEXANDER A. RAZBOROV: A simple proof of Bazzi's theorem. *ACM Transactions on Computation Theory*, 1(1):3, 2009. ECCC. [doi:10.1145/1490270.1490273] 284

[14] ROMAN SMOLENSKY: Algebraic methods in the theory of lower bounds for Boolean circuit complexity. In *Proc. 19th STOC*, pp. 77–82. ACM Press, 1987. [doi:10.1145/28395.28404] 284

[15] LUCA TREVISAN: A note on approximate counting for k-DNF. In *Proc. 8th Internat. Workshop on Randomization and Computation (RANDOM'04)*, pp. 417–425. Springer, 2004. [doi:10.1007/978-3-540-27821-4_37] 285

[16] EMANUELE VIOLA: The sum of D small-bias generators fools polynomials of degree D. *Comput. Complexity*, 18(2):209–217, 2009. Preliminary version in CCC'08. [doi:10.1007/s00037-009-0273-5] 284

[17] EMANUELE VIOLA AND AVI WIGDERSON: Norms, XOR lemmas, and lower bounds for polynomials and protocols. *Theory of Computing*, 4(7):137–168, 2008. Preliminary version in CCC'07. [doi:10.4086/toc.2008.v004a007] 285

AUTHORS

Andrej Bogdanov
Department of Computer Science and Engineering
Institute for Theoretical Computer Science and Communications
The Chinese University of Hong Kong
andrejb@cse.cuhk.edu.hk
http://www.cse.cuhk.edu.hk/~andrejb

Zeev Dvir
Department of Computer Science and Department of Mathematics
Princeton University
Princeton NJ
zeev.dvir@gmail.com
http://www.cs.princeton.edu/~zdvir

Elad Verbin
Department of Computer Science, CTIC and MADALGO
Aarhus University
Denmark
elad.verbin@gmail.com
http://www.cs.au.dk/~eladv/

Amir Yehudayoff
Department of Mathematics
Technion-IIT
Haifa, Israel
amir.yehudayoff@gmail.com
http://www.technion.ac.il/~yehuday/

ABOUT THE AUTHORS

ANDREJ BOGDANOV grew up in Macedonia in the former Yugoslavia. He tried to take his first steps in computing on a Galaksija but the machine wouldn't start. Later he attended summer camps for kids who like geometry, the Cauchy-Schwarz inequality (still a favorite), and swimming in lake Ohrid at sunset. Andrej went on to MIT, Berkeley, The Institute for Advanced Study, DIMACS at Rutgers, and ITCS at Tsinghua University, before joining, in 2008, the Chinese University of Hong Kong where he is an assistant professor at the Department of Computer Science and Engineering and associate director of the Institute of Theoretical Computer Science and Communications. His research interests include pseudorandomness, cryptography, and sublinear-time algorithms.

ZEEV DVIR is an assistant professor at Princeton University jointly appointed by the Computer science and Mathematics departments. Prior to that he was a postdoc at the Institute for Advanced Study at Princeton. He received his Ph. D. from the Weizmann Institute in Israel in 2008. His advisors were Ran Raz and Amir Shpilka.

ELAD VERBIN did his Ph. D. with Haim Kaplan (2003) at Tel Aviv University. He is currently a postdoc in the Computer Science Department of Aarhus University, jointly appointed by MADALGO and the computational complexity group.

AMIR YEHUDAYOFF did his Ph. D. at the Weizmann Institute of Science under the supervision of Ran Raz. He then spent two years as a member of the Institute for Advanced Study, Princeton, hosted by Avi Wigderson. He is currently a member of the Mathematics department at the Technion—Israel Institute of Technology. Amir also enjoys moving, dancing, and improvisation in movement.