Separation Between Read-once Oblivious Algebraic Branching Programs (ROABPs) and Multilinear Depth Three Circuits

Neeraj Kayal[1], Vineet Nair[2], and Chandan Saha[3]

1 **Microsoft Research India**
`neeraka@microsoft.com`
2 **Indian Institute of Science, India**
`vineet.nair@csa.iisc.ernet.in`
3 **Indian Institute of Science, India**
`chandan@csa.iisc.ernet.in`

Abstract

We show an exponential separation between two well-studied models of algebraic computation, namely read-once oblivious algebraic branching programs (ROABPs) and multilinear depth three circuits. In particular we show the following:

1. There exists an explicit n-variate polynomial computable by linear sized multilinear depth three circuits (with only two product gates) such that every ROABP computing it requires $2^{\Omega(n)}$ size.
2. Any multilinear depth three circuit computing $\mathrm{IMM}_{n,d}$ (the iterated matrix multiplication polynomial formed by multiplying d, $n \times n$ symbolic matrices) has $n^{\Omega(d)}$ size. $\mathrm{IMM}_{n,d}$ can be easily computed by a $\mathrm{poly}(n, d)$ sized ROABP.
3. Further, the proof of 2 yields an exponential separation between multilinear depth four and multilinear depth three circuits: There is an explicit n-variate, degree d polynomial computable by a $\mathrm{poly}(n, d)$ sized multilinear depth four circuit such that any multilinear depth three circuit computing it has size $n^{\Omega(d)}$. This improves upon the quasi-polynomial separation result by Raz and Yehudayoff [2009] between these two models.

The hard polynomial in 1 is constructed using a novel application of expander graphs in conjunction with the evaluation dimension measure used previously in Nisan [1991], Raz [2006,2009], Raz and Yehudayoff [2009], and Forbes and Shpilka [2013], while 2 is proved via a new adaptation of the dimension of the partial derivatives measure used by Nisan and Wigderson [1997]. Our lower bounds hold over any field.

1998 ACM Subject Classification F.1.3 Complexity Measures and Classes

Keywords and phrases multilinear depth three circuits, read-once oblivious algebraic branching programs, evaluation dimension, skewed partial derivatives, expander graphs, iterated matrix multiplication

Digital Object Identifier 10.4230/LIPIcs.STACS.2016.46

1 Introduction

Proving lower bounds and separating complexity classes lie at the heart of complexity theory. In algebraic complexity, separating classes VP and VNP (the algebraic analogues of P and NP) equates to proving super-polynomial lower bounds for arithmetic circuits. Another prominent and pertinent problem is polynomial identity testing (PIT). Here we are given an arithmetic circuit computing a multivariate polynomial over some field and the problem is

to determine whether the polynomial is identically zero. Polynomial time randomized PIT follows easily from [6, 33, 37]. PIT is one of the very few natural problems in BPP (in fact, in co-RP) not known to be in P. Showing arithmetic circuit lower bounds and derandomizing PIT are closely related: [17] showed that a polynomial time PIT over integers implies a super-polynomial arithmetic circuit lower bound for the family of permanent polynomials or NEXP $\not\subseteq$ P/poly. [15, 1] showed that a polynomial time blackbox PIT (meaning, we are only allowed to evaluate the circuit at points from $\mathbb{F}^n$, where n is the number of inputs and $\mathbb{F}$ the underlying field) implies exponential lower bounds for circuits computing polynomials whose coefficients can be computed in PSPACE. Conversely, [17] also showed that a super-polynomial (exponential) circuit lower bound for any family of exponential-time computable multilinear polynomials implies a sub-exponential (quasi-polynomial) time algorithm for PIT, in fact blackbox PIT, using Nisan-Wigderson generators [24] and Kaltofen's [18] polynomial factorization algorithm. [8] showed a similar connection between lower bounds and PIT for low depth circuits. They proved lower bounds for bounded depth circuits imply efficient PIT for bounded depth circuits computing polynomials with low individual degree. So, in this certain sense the complexity of proving strong lower bounds and devising efficient PIT algorithms are quite similar. Derandomizing PIT is also interesting in its own right. It is well-known that such a derandomization would imply the problem of checking existence of a perfect matching in a given graph is in NC [35].

Research over the the past several years has made notable progress on both lower bounds and PIT for interesting special cases of arithmetic circuits and helped identify the frontiers of our current knowledge. In particular, we understand better the reason why super-polynomial lower bounds and poly-time PIT have remained elusive even for depth three circuits: An exponential lower bound (similarly, a poly-time blackbox PIT) for depth three circuits over fields of characteristic zero implies an exponential lower bound (similarly, quasi-polynomial-time PIT) for general circuits [12]. For more on arithmetic circuit lower bounds and PIT refer to the surveys [34], [4], [30, 21], [31, 32].

A potentially useful and interesting restriction to consider at depth three is multilinearity (meaning, every product gate computes a multilinear polynomial). Most of the hard polynomials used in the literature are multilinear, e.g. determinant, permanent, iterated matrix multiplication, Nisan-Wigderson polynomials etc. So, it is worthwhile to develop a fuller understanding of multilinear models [27, 26, 28, 29, 7]. We do know of strong lower bounds for multilinear depth three circuits due to [29] and also this paper (Theorem 9), but as yet no efficient (meaning, quasi-polynomial) PIT is known for this model. One reason for this is the absence of hardness versus randomness tradeoff results for bounded depth multilinear circuits. Recently, [5] has given a sub-exponential time blackbox PIT algorithm for multilinear depth three circuits using recently found quasi-polynomial blackbox PIT for another model, namely read-once oblivious algebraic branching programs (ROABPs) [10, 2] (Definition 2), thereby connecting these two interesting models of computation. Could there be a more efficient reduction from multilinear depth three circuits to ROABPs? If so then that would readily imply an efficient PIT algorithm for multilinear depth three circuits. This question has lead us to this work.

Related work and motivation. The model ROABP (see Definition 2) has been studied intensely in the recent years in the context of black-box PIT, equivalently *hitting-set generators* (Definition 20). This has resulted in deterministic, quasi-polynomial time hitting-set generators for ROABPs [2, 10] and other associated models like set-multilinear algebraic branching programs [9, 10] (a special case of which is set-multilinear depth three circuits [3, 10], see

also Definition 4), non-commutative algebraic branching programs [10] and diagonal depth-3 circuits [3, 10]. Quite recently, [5] has given a $2^{\tilde{O}(n^{\frac{2}{3}(1+\delta)})}$ time hitting-set generator for multilinear depth three circuits of size at most 2^{n^δ} by 'reducing' a multilinear depth three circuit to a collection of ROABPs and 'putting together' the hitting-sets of the ROABPs. This 'putting together' process raises the hitting-set complexity from quasi-polynomial (for a single ROABP) to sub-exponential (for a composition of several ROABPs). Had it been the case that a multilinear depth three circuit can be directly reduced to a single small size ROABP, an efficient hitting set for the former would have ensued immediately from [2, 10]. One of the results in the paper (Theorem 7), rules out this possibility. In fact, Theorem 7 shows something stronger as described below.

A closer look at [2] and [5] reveals an interesting, and potentially useful, intermediate model that we call *superposition of (two or more) set-multilinear depth three circuits* (see Definition 5). An example of superposition of two set-multilinear depth three circuits is,

$$C(X,Y) = (1 + 3x_1 + 5y_2)(4 + x_2 + y_1) + (6 + 9x_1 + 4y_1)(2 + 5x_2 + 3y_2).$$

The variable sets $X = \{x_1, x_2\}$ and $Y = \{y_1, y_2\}$ are completely disjoint and are called the *base sets* of $C(X,Y)$. When projected on X variables (i.e after putting the Y variables to zero), $C(X,Y)$ is a set-multilinear depth three circuit in the X variables. A similar thing is true for the Y variables. Thus, every base set is associated with a set-multilinear depth three circuit and vice versa. Any multilinear depth three circuit can be trivially viewed as a superposition of n set-multilinear depth three circuits with single variable in every base set, where n is the number of variables. A crucial observation in [5] is that every multilinear depth three circuit is *"almost"* a superposition of n^ϵ set-multilinear depth three circuits for some $\epsilon < 1$, and further the associated n^ϵ base sets can be found in sub-exponential time using k-wise independent hash functions. Once we know the $r = n^\epsilon$ base sets corresponding to r set-multilinear depth three circuits whose superposition forms a circuit of size s, finding a hitting set for the circuit in time $s^{r.\log s}$ follows easily by taking a direct product of hitting sets for r many ROABPs (in fact, set-multilinear depth three circuits). We think a useful model to consider at this juncture is superposition of constantly many set-multilinear depth three circuits with *unknown* base sets. In this case knowing the $r = O(1)$ base sets readily gives us a quasi-polynomial time hitting set, but finding these base sets from a given circuit is NP-hard for $r \geq 3$ (as we show in Observation 6), which rules out the possibility of knowing the base-sets even if we are allowed to see the circuit (as in the *white-box* case). Indeed, even in this special case where the given multilinear depth three circuit is promised to be a superposition of constantly many (say, 2) set-multilinear depth three circuits, the algorithm in [5] finds and works with many base sets and the resulting hitting set complexity grows to roughly $\exp(\sqrt{n})$. Could it be that superposition of constantly many set-multilinear depth three circuits efficiently reduce to ROABPs? Unfortunately, the answer to this also turns out to be negative as Theorem 7 gives an explicit example of a superposition of *two* set-multilinear depth three circuit computing an n-variate polynomial f such that any ROABP computing f has width $2^{\Omega(n)}$.

While comparing two models (here multilinear depth three circuits and ROABPs), it is desirable to show a separation in both directions whenever an efficient reduction from one to the other seems infeasible. In this sense, we show a complete seperation between the models under consideration by giving an explicit polynomial computable by a polynomial sized ROABP such that every multilinear depth three circuit computing it requires exponential size. In fact, this explicit polynomial is simply the Iterated Matrix Multiplication $\text{IMM}_{n,d}$ – the $(1,1)$-th entry of a product of d $n \times n$ symbolic matrices (Theorem 9). $\text{IMM}_{n,d}$ can

be easily computed by a polynomial-sized ROABP (see Observation 10). Although, a $2^{\Omega(d)}$ lower bound for multilinear depth three circuit computing Det_d is known [29], this does not imply a lower bound for $\mathrm{IMM}_{n,d}$ (despite the fact that Det and IMM are both complete for algebraic branching programs (ABPs) [22]) as the projection from IMM to Det can make the circuit non-multilinear. Another related work by [7] showed a separation between multilinear ABPs and multilinear formulas by exhibiting an explicit polynomial (namely, an *arc-full-rank* polynomial) that is computable by a linear size multilinear ABP but requires super-polynomial size multilinear formulas. But again multilinearity of a circuit can be lost when IMM is projected to arc-full-rank polynomials, and hence this result too does not imply a lower bound for IMM. An extension of Theorem 9 to a super-polynomial lower bound for multilinear formulas computing IMM will have interesting consequences in separating noncommutative formulas and noncommutative ABPs. In a contemporary work [20], some of the authors of this work and Sébastien Tavenas have been able to show an $n^{\Omega(\sqrt{d})}$ lower bound for multilinear depth four circuits computing $\mathrm{IMM}_{n,d}$ by significantly extending a few of the ideas present in this work and building upon (thereby improving) the work of [11]. Thus, in summary the models poly-sized ROABPs and poly-sized multilinear depth three ciruits have provably different computational powers, although they share a non-trivial intersection as poly-sized set-multilinear depth three circuits is harbored in both.

An interesting outcome of the proof of the lower bound for multilinear depth three circuits computing IMM is an exponential separation between multilinear depth three and multilinear depth four circuits. Previously, [29] showed a super-polynomial separation between multilinear constant depth h and depth $h+1$ circuits, which when applied to the depth three versus four setting gives a quasi-polynomial seperation between the two models. In comparison, Theorem 11 gives an exponential separation.

The models and our results. We define the relevant models and state our results now.

▶ **Definition 1** (Algebraic Branching Program)**.** An Algebraic Branching Program(ABP) in the variables $X = \{x_1, x_2, ..., x_n\}$ is a directed acyclic graph with a source vertex s and a sink vertex t. It has $(d+1)$ sets or layers of vertices $V_1, V_2, ..., V_{d+1}$, where V_1 and V_{d+1} contain only s and t respectively. The width of an ABP is the maximum number of vertices in any of the $(d+1)$ layers. All the edges in an ABP are such that an edge starts from a vertex in V_i and is directed to a vertex in V_{i+1}, where V_i belongs to the set $\{V_1, V_2, ..., V_d\}$. The edges in an ABP are labelled by polynomials over a base field $\mathbb{F}$. The weight of a path between any two vertices u and v in an ABP is computed by taking the product of the edge labels on the path from u to v. An ABP computes the sum of the weights of all the paths from s to t.

Note that in another standard definition of an ABP, the edges are labeled by linear polynomials over a base field $\mathbb{F}$. A special kind of ABP, namely ROABP, is defined in [10].

▶ **Definition 2** (Read-Once Oblivious Algebraic Branching Program)**.** A Read-Once Oblivious Algebraic Branching Program(ROABP) over a field $\mathbb{F}$ has an associated permutation $\pi : [n] \to [n]$ of the variables in X. The number of variables is equal to the number of layers of vertices minus one, i.e. $n = (d+1) - 1 = d$. The label associated with an edge from a vertex in V_i to a vertex in V_{i+1} is an univariate polynomial over $\mathbb{F}$ in the variable $x_{\pi(i)}$.

▶ **Definition 3** (Multilinear depth 4 and depth 3 circuits)**.** A circuit $C = \sum_{i=1}^{s} \prod_{j=1}^{d_i} Q_{ij}(X_j^i)$ is a multilinear depth four ($\Sigma\Pi\Sigma\Pi$) circuit in X variables over a field $\mathbb{F}$, if $X = \uplus_{j=1}^{d_i} X_j^i$ and $Q_{ij} \in \mathbb{F}[X_j^i]$ is a multilinear polynomial for every $i \in [s]$ and $j \in [d_i]$. If Q_{ij}'s are linear polynomials then C is a multilinear depth three ($\Sigma\Pi\Sigma$) circuit. The parameter s is the *top fan-in* of C.

▶ **Definition 4** (Set-multilinear depth three circuit)**.** A circuit $C = \sum_{i=1}^{s} \prod_{j=1}^{d} l_{ij}(X_j)$ is a set-multilinear depth three ($\Sigma\Pi\Sigma$) circuit in X variables over a field $\mathbb{F}$, if $X = \uplus_{j=1}^{d} X_j$ and $l_{ij} \in \mathbb{F}[X_j]$ is a linear polynomial for every $i \in [s]$ and $j \in [d]$. The sets $X_1, X_2, ..., X_d$ are called the *colors* of X. If $|X_j| = 1$ for every $j \in [d]$ then we say X has singleton colors and C is a set-multilinear depth three circuit with singleton colors.

As a bridge between multilinear and set-multilinear depth three circuits we define a model called superposition of set-multilinear depth three circuits.

▶ **Definition 5** (Superposition of set-multilinear depth three circuits)**.** A multilinear depth three ($\Sigma\Pi\Sigma$) circuit C over a field $\mathbb{F}$ is a superposition of t set-multilinear depth three circuits over variables $X = \uplus_{i=1}^{t} Y_i$, if for every $i \in [t]$, C is a set-multilinear depth three circuit in Y_i variables over the field $\mathbb{F}(X \setminus Y_i)$. The sets $Y_1, ..., Y_t$ are called the *base sets* of C. Further, we restrict the Y_i to have singleton colors for every $i \in [t]$.

Although the notion of superposition makes sense even if Y_i's do not have singleton colors, we restrict to singletons as this model itself captures multilinear depth three circuits. We make the following observation for superposition of set-multilinear depth three circuits.

▶ **Observation 6.** *Given a circuit C which is a superposition of t set-multilinear circuits on unknown base sets $Y_1, Y_2, ..., Y_t$, finding t base sets $Y_1', Y_2', ..., Y_t'$ such that C is a superposition of t set-multilinear circuits on base sets $Y_1', Y_2', ..., Y_t'$ is NP-hard when $t > 2$.*

Due to space constraint the proof of Observation 6 is omitted. It can be found in an extended version of this paper [19]. We now state the main results of this paper.

▶ **Theorem 7** (Main Theorem 1)**.**

1. *There is an explicit family of $2n$-variate polynomials $\{g_n\}_{n\geq 1}$ over any field $\mathbb{F}$ such that the following hold: g_n is computable by a multilinear depth three circuit C over $\mathbb{F}$ with top fanin three and C is also a superposition of two set-multilinear depth three circuits. Any ROABP over $\mathbb{F}$ computing g_n has width $2^{\Omega(n)}$.*
2. *There is an explicit family of $3n$-variate polynomials $\{g_n\}_{n\geq 1}$ over any field $\mathbb{F}$ such that the following hold: g_n is computable by a multilinear depth three circuit C over $\mathbb{F}$ with top fanin two and C is also a superposition of three set-multilinear depth three circuits. Any ROABP over $\mathbb{F}$ computing f_n has width $2^{\Omega(n)}$.*

We prove Theorem 7 in Section 3. The tightness of the theorem is shown by this observation.

▶ **Observation 8.** *A polynomial computed by a multilinear $\Sigma\Pi\Sigma$ circuit with top fan-in two and at most two variables per linear polynomial can also be computed by an ROABP with constant width.*

In the interest of space the proof of Observation 8 is omitted, see [19]. Thus, it follows from Theorem 7 that if we increase either the top fan-in or the number of variables per linear polynomial from two to three in multilinear depth three circuits then there exist polynomials computed by such circuits such that ROABPs computing these polynomials have exponential width. We now state the "converse" of Theorem 7.

▶ **Theorem 9** (Main Theorem 2)**.** *Any multilinear depth three circuit (over any field) computing $\mathrm{IMM}_{n,d}$, the $(1,1)$-th entry of a product of d $n \times n$ symbolic matrices, has top fan-in $n^{\Omega(d)}$ for $n \geq 11$. (Note: This also implies a lower bound for determinant, see Corollary 38.)*

We prove Theorem 9 in Section 4. It is not hard to observe the following.

▶ **Observation 10.** *$\mathrm{IMM}_{n,d}$ can be computed by an n^2 width ROABP.*

The proof of Observation 10 is omitted, see [19]. Thus, Theorem 7, Theorem 9 and Observation 10 together imply a complete separation between multilinear depth three circuits and ROABPs. As a consequence of the proof of Theorem 9 we also get an exponential separation between multilinear depth three and multilinear depth four circuits (proof in Section 4).

▶ **Theorem 11.** *There is an explicit family of $O(n^2d)$-variate polynomials of degree d, $\{f_d\}_{d\geq 1}$, such that f_d is computable by a $O(n^2d)$-sized multilinear depth four circuit with top fan-in one (i.e. a $\Pi\Sigma\Pi$ circuit) and every multilinear depth three circuit computing f_d has top fan-in $n^{\Omega(d)}$ for $n \geq 11$.*

Observe that the hard polynomials used in Theorem 7 belong to a special class of multilinear depth three circuits – they are both superpositions of constantly many set-multilinear depth three circuits and simultaneously a sum of constantly many set-multilinear depth three circuits. Here is an example of a circuit from this class.

$$\begin{aligned} C(X,Y) = &(1+3x_1+5y_2)(4+x_2+y_1)+(9+6x_1+4y_2)(3+2x_2+5y_1) \\ &+(6+9x_1+4y_1)(2+5x_2+3y_2)+(3+6x_1+9y_1)(5+8x_2+2y_2) \end{aligned}$$

$C(X,Y)$ is a superposition of two set-multilinear depth three circuits with base sets $X = \{x_1\} \cup \{x_2\}$ and $Y = \{y_1\} \cup \{y_2\}$. But $C(X,Y)$ is also a sum of two set-multilinear depth three circuits with $\{x_1, y_2\}$, $\{x_2, y_1\}$ being the colors in the first set-multilinear depth three circuit (corresponding to the first two products) and $\{x_1, y_1\}$, $\{x_2, y_2\}$ the colors in the second set-multilinear depth three circuit (corresponding to the last two products). For such a subclass of multilinear depth three circuits, we give a quasi-polynomial time hitting set by extending the proof technique of [3].

▶ **Theorem 12.** *Let $\mathcal{C}_{n,m,l,s}$ be a subclass of multilinear depth three circuits computing n-variate polynomials such that every circuit in $\mathcal{C}_{n,m,l,s}$ is a superposition of at most m set-multilinear depth three circuits and simultaneously a sum of at most l set-multilinear depth three circuits, and has top fan-in bounded by s. There is a hitting-set generator for $\mathcal{C}_{n,m,l,s}$ running in $(ns)^{O(lm\log s)}$ time.*

When m and l are bounded by $\mathsf{poly}(\log ns)$, we get quasi-polynomial time hitting sets. The proof of Theorem 12, which extends the shift and rank concentration technique of [3], is omitted, see [19]. To our understanding, even if m and l are constants, [5]'s algorithm yields an $exp(\sqrt{n})$ hitting set complexity. Also, [13] has recently given a $(ndw)^{O(l2^l\log(ndw))}$ time hitting set generator for n-variate, individual (variable) degree d polynomials computed by sum of l ROABPs each of width less than w. Sum of l set-multilinear depth three circuits reduces to sum of l ROABPs as set-multilinear depth three circuits readily reduce to poly-sized ROABPs. But, observe the doubly exponential dependence on l in their result. On the contrary, in Theorem 12 the dependence is singly exponential in l. So, the hitting-set complexity remains quasi-polynomial for $l = (\log n)^{O(1)}$ whereas [13] gives an exponential time hitting-set generator when applied to the model in Theorem 12. However, it is also important to note that the model considered in Theorem 12 is somewhat weaker than the sum of ROABPs model in [13] because of the additional restriction that our model is also a superposition of m set-multilinear depth three circuits.

2 Preliminaries

Measures. We have used two complexity measures, namely evaluation dimension and a novel variant of the dimension of the space of partial derivatives, to prove Theorem 7 and 9

respectively. Evaluation dimension was first defined in [10][1]. Let X be a set of variables.

▶ **Definition 13** (Evaluation Dimension)**.** The evaluation dimension of a polynomial $g \in \mathbb{F}[X]$ with respect to a set $S \subseteq X$, denoted as $\text{Evaldim}_S(g)$, is defined as

$$\dim(\text{span}_{\mathbb{F}}\{g(X)|_{\forall x_j \in S \ x_j=\alpha_j} : \forall x_j \in S \ \alpha_j \in \mathbb{F}\}).$$

Evaluation dimension is a nearly equivalent variant of another measure, the *rank of the partial derivatives matrix*, defined and used earlier in [27, 26, 28, 29, 7] to prove lower bounds and separations for several multilinear models. These two measures are identical over fields of characteristic zero (or sufficiently large), but the former is well defined over any field.

The partial derivatives measure was introduced in [23, 25]. The following is a simple variant of this measure that is also inspired by the measure used in [27].

▶ **Definition 14** ("Skewed" partial derivatives)**.** Let $f \in \mathbb{F}[X, Y]$, where X and Y are disjoint sets of variables, and $\mathcal{Y}_k$ be the set of all monomials in Y variables of degree $k \in \mathbb{N}$. Define the measure $\mathsf{PD}_{\mathcal{Y}_k}(f)$ as

$$\dim\left(\text{span}_{\mathbb{F}}\left\{\left[\frac{\partial f(X,Y)}{\partial m}\right]_{\forall y \in Y \ y=0} : m \in \mathcal{Y}_k\right\}\right).$$

In proving Theorem 9, we apply the above measure with a significant difference (or *skew*) between the number of X and Y variables – it is this imbalance that plays a crucial role in the proof. It is easy to see that both the above measures obey subadditivity.

▶ **Lemma 15** (Subadditivity)**.**

1. *Let* $g_1, g_2 \in \mathbb{F}[X]$ *and* $S \subseteq X$*. Then*

$$\text{Evaldim}_S(g_1 + g_2) \leq \text{Evaldim}_S(g_1) + \text{Evaldim}_S(g_2).$$

2. *Let* $f_1, f_2 \in \mathbb{F}[X, Y]$*. Then* $\mathsf{PD}_{\mathcal{Y}_k}(f_1 + f_2) \leq \mathsf{PD}_{\mathcal{Y}_k}(f_1) + \mathsf{PD}_{\mathcal{Y}_k}(f_2)$.

Expander Graphs. A vital ingredient that helps us construct the hard polynomials in Theorem 7 is a family of explicit 3-regular expanders. We recall a few definitions from [16].

▶ **Definition 16** (Edge expansion and family of expanders)**.** Let $G = (V, E)$ be an undirected d-regular graph. For $S \subseteq V$, let $E(S, \overline{S})$ be the set of edges with one end incident on a vertex in S and the other incident on a vertex in $\overline{S} = V \backslash S$. The *edge expansion* of G denoted $h(G)$ is defined as:

$$h(G) = \min_{S:\ |S| \leq \frac{|V|}{2}} \frac{|E(S, \overline{S})|}{|S|}.$$

A sequence of d-regular graphs $\{G_i\}_{i \in \mathbb{N}}$ of size increasing with i is a *family of d-regular expanders* if there exists an $\epsilon > 0$ such that $h(G_i) > \epsilon$ for every i.

▶ **Definition 17** (Mildly explicit expanders)**.** Let $\mathcal{G} = \{G_i\}_{i \in \mathbb{N}}$ be a family of d-regular expanders such that the number of vertices in G_i is bounded by a polynomial in i. $\mathcal{G}$ is *mildly explicit* if there exists an algorithm that takes input i and constructs G_i in time polynomial in the size of G_i.

[1] They attributed the notion to Ramprasad Saptharishi.

A family of mildly explicit expanders. [16] mentions a family of mildly explicit 3-regular p-vertex expanders $\{G_p\}_{p \text{ prime}}$ such that for every graph G_p in the family: $h(G_p) > \frac{2+10^{-4}}{2}$. The vertices of G_p correspond to elements in $\mathbb{Z}_p$. A vertex x in G_p is connected to $x+1$, $x-1$ and to its inverse x^{-1} (operations are modulo p and inverse of 0 is defined as 0). We refer the reader to [16], section 11.1.2, for more details. Denote this family of 3-regular p-vertex expanders by $\mathcal{S}$.

Double Cover. The proof of Theorem 7 works with bipartite expanders. It is standard to transform a d-regular expander graph to a d-regular bipartite expander graph by taking its double cover.

▶ **Definition 18** (Double Cover)**.** The double cover of a graph $G = (V, E)$ is the bipartite graph $\tilde{G} = (L \uplus R, \tilde{E})$ where $|L| = |R| = |V|$. Corresponding to a vertex $u \in V$ we have two vertices $u_L \in L$ and $u_R \in R$. Edges (u_L, v_R) and $(u_R, v_L) \in \tilde{E}$ if and only if there is an edge $(u, v) \in E$.

▶ **Lemma 19.** *Let $\mathcal{S} = \{G_p\}_{p \text{ prime}}$ be the family of expanders as described above, and $\tilde{\mathcal{S}} = \{\tilde{G_p}\}_p$ the family of double covers of graphs in $\mathcal{S}$. Then $h(\tilde{G_p}) > \frac{2+10^{-4}}{2}$ for every p.* [In the interest of space the proof is omitted.]

Hitting-set generators. In Theorem 12, we give a quasi-polynomial time hitting-set generator for a subclass of multilinear depth three circuits.

▶ **Definition 20** (Hitting-set generators)**.** A hitting-set generator for a class of circuits $\mathcal{C}$ is a Turing machine $\mathcal{H}$ that takes $(1^n, 1^s)$ as input and outputs a set $\{a_1, \ldots, a_m\} \subseteq \mathbb{Z}^n$ such that for every circuit $C \in \mathcal{C}$ of size bounded by s and computing a nonzero n-variate polynomial over a field $\mathbb{F} \supset \mathbb{Z}$, there is an $i \in [m]$ for which $C(a_i) \neq 0$. Complexity of $\mathcal{H}$ is its running time. Hitting-set generators can be defined similarly over finite fields by considering field extensions.1

Technical Lemmas. The following lemmas are used in Theorem 7. Lemma 21 follows from Hall's marriage theorem [14]. The proofs of Lemmas 22 and 23 are omitted, see [19].

▶ **Lemma 21.** *A d-regular graph can be split into d edge disjoint perfect matchings.*

▶ **Lemma 22.** *Suppose $g_1(X), g_2(X), ..., g_m(X) \in \mathbb{F}[X]$ are $\mathbb{F}$-linearly independent polynomials in the variables $X = \{x_1, x_2, ..., x_n\}$ where $m = 2^n$. If $Y = \{y_1, y_2, ..., y_n\}$ are n variables different from X then (by identifying an $i \in [m]$ with an $S \subseteq [n]$),*

$$\text{Evaldim}_Y\Big(\sum_{S \subseteq [n]} y_S \cdot g_S(X)\Big) = m, \quad \text{where for } S \subseteq [n],\ y_S := \prod_{j \in S} y_j.$$

▶ **Lemma 23.** *If R is a width-k ROABP that computes $g(X)$ then for every $i \in [0, |X|]$ there exists a set $S \subseteq X$ of size i such that* $\text{Evaldim}_S(g) \leq k$.

3 Lower bounds for ROABP: Proof of Theorem 7

Proof of part 1

Construction of the polynomial family. We construct a family of $2n$-variate multilinear polynomials $\{g_n\}_{n \geq 1}$ from the explicit family of 3-regular expander graphs $\mathcal{S}$ (described

in Section 2). From an n-vertex graph $G = (V, E)$ in $\mathcal{S}$, construct a polynomial $g(X, Y)$ in variables $X = \{x_1, \ldots, x_n\}$ and $Y = \{y_1, \ldots, y_n\}$ as follows: Let $\tilde{G} = (L \uplus R, \tilde{E})$ be the double cover of G. By Lemma 19, $h(\tilde{G}) > \frac{2+10^{-4}}{2}$. With every vertex in L (similarly, R) associate a unique variable in X (respectively, Y), thus vertices in L and R are identified with the X and Y variables respectively. An edge between x_i and y_j is associated with the linear polynomial $(1 + x_i + y_j)$. By Lemma 21, $\tilde{G}$ can be split into three edge disjoint perfect matchings. Polynomial $g(X, Y)$ is a sum of three product terms corresponding to the three edge disjoint perfect matchings of $\tilde{G}$; a product term is formed by taking product of the linear polynomials associated with the edges of the corresponding matching. It is easy to show the following claim. We leave the proof to the reader.

▶ **Claim 24.** *Polynomial g (constructed above) is computed by a multilinear depth three circuit C of size $\Theta(n)$ and top fan-in three, and C is a superposition of two set-multilinear depth three circuits.*

High evaluation dimension of $g(X, Y)$. It turns out that the evaluation dimension of $g(X, Y)$ with respect to any subset of variables of size $n/10$ is large.

▶ **Lemma 25.** *For any set $S \subseteq X \uplus Y$ of size $n/10$, $\text{Evaldim}_S(g) \geq 2^{\epsilon n}$ where $\epsilon > 0$ is a constant.*

Proof. Consider any subset S of $n/10$ variables from $X \uplus Y$. With respect to set S we can classify the linear polynomials in the product terms of $g(X, Y)$ into three types: *untouched* – if none of the two variables in the linear polynomial belong to S, *partially touched* – if exactly one of the variables in the linear polynomial belongs to S, and *completely touched* – if both variables belong to S. Call the three product terms of g – P_1, P_2 and P_3. Proof of the next claim is omitted, see [19].

▶ **Claim 26.** *There exists a set $X_0 \subseteq X$ of $\left(\frac{7n}{10} - 4\right)$ X-variables such that every $x \in X_0$ appears in an untouched linear polynomial in every P_i (for $i \in [3]$), and further if $(1 + x + y_{j_1}), (1 + x + y_{j_2})$ and $(1 + x + y_{j_3})$ are the linear polynomials occurring in P_1, P_2 and P_3 respectively then $y_{j_1} \neq y_{j_2} \neq y_{j_3}$.*

For $i \in [3]$, let B_i be the set of partially touched linear polynomials in term P_i.

▶ **Claim 27.** *There is an $i \in [3]$ such that $|B_i| \geq \epsilon n$ where $\epsilon = 0.01$.*

Proof. Let T be such that, for all $i \in [3]$, $|B_i| \leq T$. Recall that g has been constructed from the bipartite expander $\tilde{G}$, and vertices in $\tilde{G}$ identified with the variable set $X \uplus Y$. We denote the vertices in $\tilde{G}$ corresponding to the variables in S also by S, and denote the set of edges going out from S to $\overline{S} = L \uplus R \backslash S$ in $\tilde{G}$ by $\tilde{E}(S, \overline{S})$. Using the expansion property of $\tilde{G}$,

$$|\tilde{E}(S, \overline{S})| \geq h(\tilde{G}) \cdot |S| \geq \frac{2 + 10^{-4}}{2} \cdot \left(\frac{n}{10}\right).$$

Every edge in $\tilde{E}(S, \overline{S})$ corresponds to a partially touched linear polynomial. Since $\tilde{G}$ is 3-regular, at least $\frac{|\tilde{E}(S,\overline{S})|}{3}$ of the edges correspond to distinct partially touched linear polynomials. By assumption, the number of such partially touched linear polynomials is at most $3T$; and so $T \geq 0.01n$. ◀

The next claim completes the proof of Lemma 25.

▶ **Claim 28.** *If there exists an $i \in [3]$ such that $|B_i| \geq \epsilon n$ for $\epsilon > 0$, then $\text{Evaldim}_S(g) \geq 2^{\epsilon n}$.*

Proof. Without loss of generality, assume $|B_1| \geq \epsilon n$. Pick two variables, say x and x', from the set X_0 (as described in Claim 26). Let $(1 + x + y_{j_2})$ and $(1 + x' + y'_{j_3})$ be the linear polynomials appearing in P_2 and P_3 respectively. By substituting $x = -(1 + y_{j_2})$ and $x' = -(1 + y'_{j_3})$ in g, the terms P_2 and P_3 vanish but P_1 does not (by Claim 26). Let $\hat{g}$ be the polynomial g after the substitution. Polynomial $\hat{g}$ has only one product term $\hat{P}_1$ (i.e. P_1 under the substitution), and $\hat{P}_1$ has as many partially touched linear polynomials as P_1. At this point, it is not difficult to prove the following observation.

▶ **Observation 29.** $\text{Evaldim}_S(g) \geq \text{Evaldim}_S(\hat{g}) = \text{Evaldim}_S(\hat{P}_1) \geq 2^{\epsilon n}$.

This completes the proof of Claim 28. ◀

From Lemma 23 and 25 we conclude that any ROABP computing $g(X, Y)$ has width at least $2^{\epsilon n}$. ◀

Proof of part 2

Construction of the polynomial family. Similar to part 1, we construct a family of $3n$-variate multilinear polynomials $\{g_n\}_{n\geq 1}$ from the explicit family of 3-regular expanders $\mathcal{S}$ – but this time edges will be associated with variables and vertices with linear polynomials. From an n-vertex graph $G = (V, E)$ in $\mathcal{S}$, construct a polynomial $g(X, Y, Z)$ in variables $X = \{x_1, \ldots, x_n\}$, $Y = \{y_1, \ldots, y_n\}$ and $Z = \{z_1, \ldots, z_n\}$ as follows: Let $\tilde{G} = (L \uplus R, \tilde{E})$ be the double cover of G, and as before $h(\tilde{G}) > \frac{2+10^{-4}}{2}$. Edges of $\tilde{G}$ can be split into three edge disjoint perfect matchings (by Lemma 21). Label the edges of the first perfect matching by distinct X-variables, the edges of the second matching by distinct Y-variables, and the edges of the third by distinct Z-variables. Vertices of $\tilde{G}$ now correspond to linear polynomials naturally – if the three edges incident on a vertex are labelled x_i, y_j and z_k then associate the linear polynomial $(1 + x_i + y_j + z_k)$ with the vertex. Let P_1 be the product of the linear polynomials associated with the vertices of L, and P_2 the product of linear polynomials associated with the vertices of R. Polynomial $g(X, Y, Z)$ is the sum of P_1 and P_2. The following claim is easy to show (just like Claim 24).

▶ **Claim 30.** *Polynomial g (constructed above) is computed by a multilinear depth three circuit C of size $\Theta(n)$ and top fan-in two, and C is a superposition of three set-multilinear depth three circuits.*

High evaluation dimension of $g(X, Y)$. The proof of the following lemma is similar to that of Lemma 25, differences arise only due to the 'dual' nature of g.

▶ **Lemma 31.** *For any $S \subseteq X \uplus Y \uplus Z$ of size $n/10$, $\text{Evaldim}_S(g) \geq 2^{\epsilon n}$ where $\epsilon > 0$ is a constant.*

Proof. Let S be any set of $\frac{n}{10}$ variables from $X \uplus Y \uplus Z$. The definitions of untouched, partially touched and completely touched linear polynomials are almost the same as in the proof of Lemma 25. The difference is we have three variables instead of two in a linear polynomial in g. So, a linear polynomial is partially touched if at most two of the three variables belong to S. For $i \in [2]$, let B_i be the set of partially touched linear polynomials and C_i the set of completely touched linear polynomials in product term P_i of g.

▶ **Claim 32.** *There is an $i \in [2]$ such that $|B_i| \geq \epsilon n$ where $\epsilon = 0.01$.*

Proof. Let T be such that, for all $i \in [2]$, $|B_i| \leq T$. The proof of the next observation is omitted, see [19].

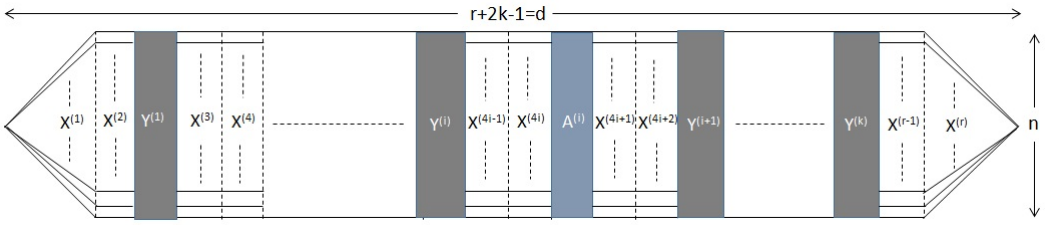

Figure 1 ABP $\mathcal{M}$.

▶ **Observation 33.** *$|C_1| + |C_2|$ is at least $\frac{n}{15} - \frac{8T}{3}$.*

Let C be the set of vertices in $\tilde{G}$ corresponding to the completely touched linear polynomials in both the product gates, thus $|C| = |C_1| + |C_2| \geq \frac{n}{15} - \frac{8T}{3}$. Each edge in $\tilde{E}(C, \overline{C})$ connects a vertex which corresponds to a completely touched linear polynomial to a vertex which corresponds to a partially touched linear polynomial. Using expansion of $\tilde{G}$,

$$|\tilde{E}(C, \overline{C})| \geq h(\tilde{G}) \cdot |C| \geq \frac{2 + 10^{-4}}{2} \cdot \left(\frac{n}{15} - \frac{8T}{3} \right).$$

Since edges in $\tilde{E}(C, \overline{C})$ are associated with variables in S, a vertex corresponding to a partially touched linear polynomial has at most two edges from $\tilde{E}(C, \overline{C})$ incident on it. Hence the number of vertices corresponding to partially touched linear polynomials is at least $\frac{\tilde{E}(C,\overline{C})}{2}$. But, by assumption, the number of such vertices is at most $2T$. Thus,

$$2T \geq \frac{|\tilde{E}(C, \overline{C})|}{2} \geq \frac{2 + 10^{-4}}{4} \cdot \left(\frac{n}{15} - \frac{8T}{3} \right) \quad \Rightarrow \quad T \geq 0.01n\,.$$

◀

The proof of the next claim is much like that of Claim 28 and is omitted.

▶ **Claim 34.** *If there exists an $i \in [2]$ such that $|B_i| \geq \epsilon n$ for $\epsilon > 0$, then* $\text{Evaldim}_S(g) \geq 2^{\epsilon n}$.

This completes the proof of Lemma 31. From Lemma 23 and 31 we conclude that any ROABP computing g has width at least $2^{\epsilon n}$. ◀

4 Lower bounds for multilinear depth three circuits

The proofs of Theorems 9 and 11 are inspired by a particular kind of *projection* of $\text{IMM}_{n,d}$ considered in [11]. We say a polynomial f is a *simple projection* of another polynomial g if f is obtained by simply setting some variables to field constants in g.

Proof of Theorem 9. The proof proceeds by constructing an ABP $\mathcal{M}$ of width n and with $d + 1$ layers of vertices such that (a) the polynomial computed by $\mathcal{M}$, say f, is a simple projection of $\text{IMM}_{n,d}$, and (b) any multilinear depth three circuit computing f has top fan-in $n^{\Omega(d)}$. Since an ABP can be viewed equivalently as a product of matrices, we will describe $\mathcal{M}$ using matrices. Figure 1 depicts the ABP $\mathcal{M}$.

Description of $\mathcal{M}$. The polynomial f, computed by $\mathcal{M}$, is defined over two disjoint sets of variables, X and Y. The Y variables are contained in k matrices, $\{Y^{(1)}, ..., Y^{(k)}\}$; the (u, v)-th entry in $Y^{(i)}$ is a formal variable $y^{(i)}_{u,v}$. There are $(k-1)$ matrices $\{A^{(1)}, ..., A^{(k-1)}\}$, such that all the entries in these matrices are ones. The X variables are contained in r matrices, $\{X^{(1)}, ..., X^{(r)}\}$. Matrices $X^{(1)}$ and $X^{(r)}$ are row and column vectors of size n respectively. The u-th entry in $X^{(1)}$ (similarly, $X^{(r)}$) is $x^{(1)}_u$ (respectively, $x^{(r)}_u$). All the remaining matrices $\{X^{(2)}, ..., X^{(r-1)}\}$ are diagonal matrices in the X variables, i.e. the (u, u)-th entry in $X^{(i)}$ is $x^{(i)}_u$ and all other entries are zero for $i \in [2, r-1]$. The matrices are placed as follows: Between two adjacent Y matrices, $Y^{(i)}$ and $Y^{(i+1)}$, we have five matrices ordered from left to right as $X^{(4i-1)}, X^{(4i)}, A^{(i)}, X^{(4i+1)}$ and $X^{(4i+2)}$ for every $i \in [1, k-1]$. Ordered from left to right, $X^{(1)}$ and $X^{(2)}$ are on the left of $Y^{(1)}$ and $X^{(r-1)}$ and $X^{(r)}$ are on the right of $Y^{(k)}$. Naturally, we have the following relation among k, r and d: $r = 4k$ and $d = r + 2k - 1$, i.e. $k = \frac{d+1}{6}$. Thus $|X| = nr = 4nk$ and $|Y| = n^2k$. This imbalance between the X and Y variables plays a vital role in the proof. As before, call the polynomial computed by this ABP $\mathcal{M}$ as $f(X, Y)$.

The following claim is easy to verify as f is a simple projection of $IMM_{n,d}$.

▶ **Claim 35.** *If* $\mathrm{IMM}_{n,d}$ *is computed by a multilinear depth three circuit having top fan-in* s *then* f *is also computed by a multilinear depth three circuit having top fan-in* s*.*

We show every multilinear depth three circuit computing f has top fan-in $n^{\Omega(d)}$ for $n \geq 11$.

Lower bounding $\mathsf{PD}_{\mathcal{Y}_k}(f)$. Let $\tilde{\mathcal{Y}}_k \subseteq \mathcal{Y}_k$ be the set of monomials formed by picking exactly one Y-variable from each of the matrices $Y^{(1)}, ..., Y^{(k)}$ and taking their product. Then, $|\tilde{\mathcal{Y}}_k| = n^{2k}$.

▶ **Claim 36.** $\mathsf{PD}_{\mathcal{Y}_k}(f(X, Y)) = |\tilde{\mathcal{Y}}_k| = n^{2k}$.

Proof. The derivative of f with respect to a monomial $m \in \mathcal{Y}_k$ is nonzero if and only if $m \in \tilde{\mathcal{Y}}_k$. Also, such a derivative $\frac{\partial f}{\partial m}$ is a multilinear degree-r monomial in X-variables. The derivatives of f with respect to two distinct monomials m and m' in $\tilde{\mathcal{Y}}_k$ give two distinct multilinear degree-r monomials in X-variables. Hence, $\mathsf{PD}_{\mathcal{Y}_k}(f) = |\tilde{\mathcal{Y}}_k|$. ◀

Upper bounding $\mathsf{PD}_{\mathcal{Y}_k}$ of a multilinear depth three circuit.

▶ **Lemma 37.** *Let* C *be a multilinear depth three circuit having top fan-in* s *computing a polynomial in* X *and* Y *variables. Then* $\mathsf{PD}_{\mathcal{Y}_k}(C) \leq s \cdot (k+1) \cdot \binom{|X|}{k}$ *if* $k \leq \frac{|X|}{2}$*.*

Proof. Let $C = \sum_{i=1}^{s} T_i$, where each T_i is a product of linear polynomials on disjoint sets of variables. From Lemma 15, $\mathsf{PD}_{\mathcal{Y}_k}(C) \leq s \cdot \max_{i \in [s]} \mathsf{PD}_{\mathcal{Y}_k}(T_i)$. We need to upper bound the dimension of the "skewed" partial derivatives of a term $T_i = T$ (say). Let $T = \prod_{j=1}^{q} l_j$, where l_j is a linear polynomial. Among the q linear polynomials at most $|X|$ of them contain the X variables. Without loss of generality, assume the linear polynomials $l_1, \ldots, l_p$ contain X-variables and the remaining $l_{p+1}, \ldots, l_q$ are X-free (here $p \leq |X|$). Let $Q = \prod_{j=p+1}^{q} l_j$. Then, $T = Q \cdot \prod_{j=1}^{p} l_j$. We take the derivative of T with respect to a monomial $m \in \mathcal{Y}_k$ and then substitute the Y variables to zero. Applying the product rule of differentiation and observing that the derivative of a linear polynomial with respect to a variable makes it a constant we have the following:

$$\left[\frac{\partial T}{\partial m}\right]_{Y=\overline{0}} = \sum_{\substack{S \subseteq [p] \\ |S| \leq k}} \alpha_S \prod_{j \in [p] \setminus S} [l_j]_{Y=\overline{0}}$$

where α_S's are constants from the field. Here m is a representative element of the set $\mathcal{Y}_k$. Hence every such derivative can be expressed as a linear combination of $\sum_{t=0}^{k}\binom{p}{t} \leq (k+1)\cdot\binom{|X|}{k}$ polynomials, where the last inequality is due to $k \leq \frac{|X|}{2}$ (if $t > p$ then $\binom{p}{t} \stackrel{\text{def}}{=} 0$). Therefore, $\mathsf{PD}_{\mathcal{Y}_k}(T) \leq (k+1)\cdot\binom{|X|}{k}$ and $\mathsf{PD}_{\mathcal{Y}_k}(C) \leq s\cdot(k+1)\cdot\binom{|X|}{k}$. ◀

It follows from Claim 36 and Lemma 37 that the top fan-in s of any multilinear depth three circuit computing $f(X,Y)$ is such that

$$s \geq \frac{n^{2k}}{(k+1)\cdot\binom{4nk}{k}} \geq \frac{n^{2k}}{(k+1)\cdot(4ne)^k} = n^{\Omega(d)},$$

as $n \geq 11$ and $k \leq |X|/2$ (required in Lemma 37). Claim 35 now completes the proof of Theorem 9. ◀

Theorem 9 implies the following corollary (already known due to [29]) as $IMM_{n,d}$ is a simple projection of Det_{nd}, the determinant of an $nd \times nd$ symbolic matrix [36].

▶ **Corollary 38** ([29]). *Any multilinear depth three circuit (over any field) computing* Det_d*, the determinant of a* $d \times d$ *symbolic matrix, has top fan-in* $2^{\Omega(d)}$*.*

Proof of Theorem 11. We now show that the polynomial $f(X,Y)$, computed by the ABP $\mathcal{M}$, can also be computed a multilinear depth four circuit of size $O(n^2d)$ and having top fan-in just one. ABP $\mathcal{M}$ has k matrices, $Y^{(1)},\ldots,Y^{(k)}$, containing the Y-variables. Associate with each matrix $Y^{(i)}$ four matrices containing the X-variables, two on the immediate left $X^{(4i-3)}$ and $X^{(4i-2)}$, and two on the immediate right $X^{(4i-1)}$ and $X^{(4i)}$. Every monomial in f is formed by picking exactly one variable from every matrix and taking their product. Once we pick $y^{(i)}_{u,v}$ from $Y^{(i)}$, this automatically fixes the variables picked from $X^{(4i-3)}, X^{(4i-2)}, X^{(4i-1)}$ and $X^{(4i)}$, as these are diagonal matrices. Moreover, any variable can be picked from $Y^{(i)}$ irrespective of which other Y-variables are picked from $Y^{(1)},\ldots,Y^{(i-1)},Y^{(i+1)},\ldots,Y^{(k)}$. This observation can be easily formalized to show that

$$f = \prod_{i=1}^{k}\sum_{u,v\in[n]} x_u^{(4i-3)}x_u^{(4i-2)}\cdot y^{(i)}_{u,v}\cdot x_v^{(4i-1)}x_v^{(4i)}.$$

The size of this multilinear ΠΣΠ circuit is $O(n^2k) = O(n^2d)$. ◀

Acknowledgements. VN and CS would like to thank Rohit Gurjar for helpful discussions on Observations 6 and 8. Thanks also to Nitin Saxena and Arpita Korwar for some early discussions at the onset of this work. NK and CS would like to thank Sébastien Tavenas for attending a presentation of this work and giving us some useful feedback. We also thank the anonymous reviewers for their helpful comments.

References

1 Manindra Agrawal. Proving lower bounds via pseudo-random generators. In *FSTTCS 2005: Foundations of Software Technology and Theoretical Computer Science, 25th International Conference, Hyderabad, India, December 15-18, 2005, Proceedings*, pages 92–105, 2005. doi:10.1007/11590156_6.

2 Manindra Agrawal, Rohit Gurjar, Arpita Korwar, and Nitin Saxena. Hitting-sets for ROABP and sum of set-multilinear circuits. *SIAM J. Comput.*, 44(3):669–697, 2015. doi:10.1137/140975103.

3 Manindra Agrawal, Chandan Saha, and Nitin Saxena. Quasi-polynomial hitting-set for set-depth-Δ formulas. In *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*, pages 321–330, 2013.

4 Xi Chen, Neeraj Kayal, and Avi Wigderson. Partial Derivatives in Arithmetic Complexity and Beyond. *Foundations and Trends in Theoretical Computer Science*, 6(1-2):1–138, 2011.

5 Rafael Mendes de Oliveira, Amir Shpilka, and Ben Lee Volk. Subexponential size hitting sets for bounded depth multilinear formulas. In *30th Conference on Computational Complexity, CCC 2015, June 17-19, 2015, Portland, Oregon, USA*, pages 304–322, 2015. doi:10.4230/LIPIcs.CCC.2015.304.

6 Richard A. DeMillo and Richard J. Lipton. A probabilistic remark on algebraic program testing. *Inf. Process. Lett.*, 7(4):193–195, 1978. doi:10.1016/0020-0190(78)90067-4.

7 Zeev Dvir, Guillaume Malod, Sylvain Perifel, and Amir Yehudayoff. Separating multilinear branching programs and formulas. In *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, New York, NY, USA, May 19-22, 2012*, pages 615–624, 2012. doi:10.1145/2213977.2214034.

8 Zeev Dvir, Amir Shpilka, and Amir Yehudayoff. Hardness-randomness tradeoffs for bounded depth arithmetic circuits. *SIAM J. Comput.*, 39(4):1279–1293, 2009. doi:10.1137/080735850.

9 Michael A. Forbes, Ramprasad Saptharishi, and Amir Shpilka. Hitting sets for multilinear read-once algebraic branching programs, in any order. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 – June 03, 2014*, pages 867–875, 2014.

10 Michael A. Forbes and Amir Shpilka. Quasipolynomial-time identity testing of noncommutative and read-once oblivious algebraic branching programs. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 243–252, 2013. doi:10.1109/FOCS.2013.34.

11 Hervé Fournier, Nutan Limaye, Guillaume Malod, and Srikanth Srinivasan. Lower bounds for depth 4 formulas computing iterated matrix multiplication. In *STOC*, pages 128–135, 2014.

12 Ankit Gupta, Pritish Kamath, Neeraj Kayal, and Ramprasad Saptharishi. Arithmetic circuits: A chasm at depth three. In *Foundations of Computer Science (FOCS)*, pages 578–587, 2013.

13 Rohit Gurjar, Arpita Korwar, Nitin Saxena, and Thomas Thierauf. Deterministic identity testing for sum of read-once oblivious arithmetic branching programs. In *30th Conference on Computational Complexity, CCC 2015, June 17-19, 2015, Portland, Oregon, USA*, pages 323–346, 2015. doi:10.4230/LIPIcs.CCC.2015.323.

14 Philip Hall. On representatives of subsets. *J. London Math. Soc.*, 10(1):26–30, 1935.

15 Joos Heintz and Claus-Peter Schnorr. Testing polynomials which are easy to compute (extended abstract). In *Proceedings of the 12th Annual ACM Symposium on Theory of Computing, April 28-30, 1980, Los Angeles, California, USA*, pages 262–272, 1980. doi:10.1145/800141.804674.

16 Shlomo Hoory, Nathan Linial, and Avi Wigderson. Expander graphs and their applications. *Bulletin of the American Mathematical Society*, 43(4):439–561, 2006.

17 Valentine Kabanets and Russell Impagliazzo. Derandomizing polynomial identity tests means proving circuit lower bounds. *Computational Complexity*, 13(1-2):1–46, 2004. doi:10.1007/s00037-004-0182-6.

18 Erich Kaltofen. Factorization of Polynomials Given by Straight-Line Programs. In *Randomness and Computation*, pages 375–412. JAI Press, 1989.

19 Neeraj Kayal, Vineet Nair, and Chandan Saha. Separation between Read-once Oblivious Algebraic Branching Programs (ROABPs) and Multilinear Depth Three Circuits.

Electronic Colloquium on Computational Complexity (ECCC), 22:154, 2015. URL: http://eccc.hpi-web.de/report/2015/154.

20 Neeraj Kayal, Chandan Saha, and Sébastien Tavenas. On the size of homogeneous and of depth four formulas with low individual degree. *Electronic Colloquium on Computational Complexity (ECCC)*, 22:181, 2015. URL: http://eccc.hpi-web.de/report/2015/181.

21 Neeraj Kayal and Ramprasad Saptharishi. A selection of lower bounds for arithmetic circuits. *Perspectives in Computational Complexity*, 2014.

22 Meena Mahajan and V. Vinay. Determinant: Combinatorics, algorithms, and complexity. *Chicago J. Theor. Comput. Sci.*, 1997, 1997. URL: http://cjtcs.cs.uchicago.edu/articles/1997/5/contents.html.

23 Noam Nisan. Lower bounds for non-commutative computation (extended abstract). In *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing, May 5-8, 1991, New Orleans, Louisiana, USA*, pages 410–418, 1991. doi:10.1145/103418.103462.

24 Noam Nisan and Avi Wigderson. Hardness vs randomness. *J. Comput. Syst. Sci.*, 49(2):149–167, 1994.

25 Noam Nisan and Avi Wigderson. Lower bounds on arithmetic circuits via partial derivatives. *Computational Complexity*, 6(3):217–234, 1997. Available at http://www.math.ias.edu/~avi/PUBLICATIONS/MYPAPERS/NW96/final.pdf.

26 Ran Raz. Separation of multilinear circuit and formula size. *Theory of Computing*, 2(1):121–135, 2006.

27 Ran Raz. Multi-linear formulas for permanent and determinant are of super-polynomial size. *J. ACM*, 56(2), 2009.

28 Ran Raz and Amir Yehudayoff. Balancing syntactically multilinear arithmetic circuits. *Computational Complexity*, 17(4):515–535, 2008.

29 Ran Raz and Amir Yehudayoff. Lower bounds and separations for constant depth multilinear circuits. *Computational Complexity*, 18(2):171–207, 2009.

30 Ramprasad Saptharishi. Recent progress on arithmetic circuit lower bounds. *Bulletin of the EATCS*, 114, 2014.

31 Nitin Saxena. Progress on polynomial identity testing. *Bulletin of the EATCS*, 99:49–79, 2009.

32 Nitin Saxena. Progress on polynomial identity testing – II. *Electronic Colloquium on Computational Complexity (ECCC)*, 20:186, 2013.

33 Jacob T. Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *J. ACM*, 27(4):701–717, 1980. doi:10.1145/322217.322225.

34 Amir Shpilka and Amir Yehudayoff. Arithmetic circuits: A survey of recent results and open questions. *Foundations and Trends in Theoretical Computer Science*, 5:207–388, March 2010. doi:10.1561/0400000039.

35 W.T. Tutte. The Factorization of Linear Graphs. *J. London Math. Soc.*, 22:107–111, 1947.

36 L. G. Valiant. Completeness Classes in Algebra. In *STOC'79: Proceedings of the eleventh annual ACM symposium on Theory of computing*, pages 249–261, New York, NY, USA, 1979. ACM Press.

37 Richard Zippel. Probabilistic algorithms for sparse polynomials. In *Symbolic and Algebraic Computation, EUROSAM'79, An International Symposiumon Symbolic and Algebraic Computation, Marseille, France, June 1979, Proceedings*, pages 216–226, 1979. doi:10.1007/3-540-09519-5_73.