Lower Bounds for Multilinear Order-Restricted ABPs

C. Ramya
Chennai Mathematical Institute, Chennai, India
ramya@cse.iitm.ac.in

B. V. Raghavendra Rao
IIT Madras, Chennai, India
bvrr@cse.iitm.ac.in

Abstract

Proving super-polynomial lower bounds on the size of syntactic multilinear Algebraic Branching Programs (smABPs) computing an explicit polynomial is a challenging problem in Algebraic Complexity Theory. The order in which variables in $\{x_1, \ldots, x_n\}$ appear along any source to sink path in an smABP can be viewed as a permutation in S_n. In this article, we consider the following special classes of smABPs where the order of occurrence of variables along a source to sink path is restricted:

1. **Strict circular-interval ABPs:** For every sub-program the index set of variables occurring in it is contained in some circular interval of $\{1, \ldots, n\}$.
2. **$\mathcal{L}$-ordered ABPs:** There is a set of $\mathcal{L}$ permutations (orders) of variables such that every source to sink path in the smABP reads variables in one of these $\mathcal{L}$ orders, where $\mathcal{L} \leq 2^{n^{1/2-\epsilon}}$ for some $\epsilon > 0$.

We prove exponential (i.e., $2^{\Omega(n^\delta)}$, $\delta > 0$) lower bounds on the size of above models computing an explicit multilinear $2n$-variate polynomial in VP.

As a main ingredient in our lower bounds, we show that any polynomial that can be computed by an smABP of size S, can be written as a sum of $O(S)$ many multilinear polynomials where each summand is a product of two polynomials in at most $2n/3$ variables, computable by smABPs. As a corollary, we show that any size S syntactic multilinear ABP can be transformed into a size $S^{O(\sqrt{n})}$ depth four syntactic multilinear $\Sigma\Pi\Sigma\Pi$ circuit where the bottom Σ gates compute polynomials on at most $O(\sqrt{n})$ variables.

Finally, we compare the above models with other standard models for computing multilinear polynomials.

2012 ACM Subject Classification Theory of computation → Algebraic complexity theory; Theory of computation → Circuit complexity

Keywords and phrases Computational complexity, Algebraic complexity theory, Polynomials

Digital Object Identifier 10.4230/LIPIcs.MFCS.2019.52

Related Version A full version of the paper is available at [17], `https://arxiv.org/abs/1901.04377`.

1 Introduction

Algebraic Complexity Theory is concerned with the classification of polynomials based on the number of arithmetic operations required to compute a polynomial from variables and constants. Arithmetic circuits are standard models for algebraic computation. One of the primary tasks in Algebraic Complexity Theory is to prove lower bounds on the size of arithmetic circuits computing an explicit polynomial. Valiant [23] conjectured that the polynomial defined by the permanent of an $n \times n$ symbolic matrix is not computable by polynomial size arithmetic circuits. This is known as Valiant's hypothesis and is one of the central questions in Algebraic Complexity Theory.

44th International Symposium on Mathematical Foundations of Computer Science (MFCS 2019).
Editors: Peter Rossmanith, Pinar Heggernes, and Joost-Pieter Katoen; Article No. 52; pp. 52:1–52:14
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

The best known size lower bound for general classes of arithmetic circuits is only super-linear in the number of variables [5]. Despite several approaches, the problem of proving lower bounds for general classes of arithmetic circuits has remained elusive. Naturally, there have been efforts to prove lower bounds for special classes of arithmetic circuits which led to the development of several lower bound techniques. Structural restrictions such as depth and fan-out, semantic restrictions such as multilinearity and homogeneity have received widespread attention in the literature.

Agrawal and Vinay [1] showed that proving exponential lower bounds for depth four circuits is sufficient to prove Valiant's hypothesis. This initiated several attempts at proving lower bounds for constant depth circuits. (See [21] for a detailed survey of these results.)

Among other restrictions, *multilinear circuits* where every gate computes a multilinear polynomial are well studied. Multilinear circuits are natural models for computing multilinear polynomials. It is sometimes more useful to consider a natural syntactic sub-class of multilinear circuits. A circuit is *syntactic multilinear* if the children of every product gate depend on disjoint sets of variables. Raz [18] obtained super-polynomial lower bounds on the size of syntactic multilinear formulas computing the determinant or permanent polynomial. This was improved to an exponential lower bound for constant-depth multilinear circuits [20, 7, 6]. However, the best known lower bound for general syntactic multilinear circuits is almost quadratic in the number of variables [2].

Algebraic Branching Programs (ABPs) are special classes of arithmetic circuits that have been studied extensively in the past. Nisan [15] obtained an exact complexity characterization of ABPs in the non-commutative setting. The problem of proving size lower bounds for the class of algebraic branching programs is widely open, even with restrictions such as homogeneity or syntactic multilinearity. When the ABP is restricted to be *homogeneous*, the best known lower bound is only quadratic in the number of variables [13]. The situation is not better in the case of syntactic multilinear ABPs, where no super-quadratic lower bound is known [2].

Models and results. In this article, we are interested in syntactic multilinear ABPs in which the order of appearance of variables along any path in the ABP is restricted. To begin with, we give a decomposition theorem for smABPs. The decomposition obtains two disjoint sets E_1 and E_2 of edges in the branching program P with source s and sink t such that the polynomial computed by P can be expressed as a sum of $\sum_{(u,v)\in E_1}[s,u]\cdot\mathsf{label}(u,v)\cdot[v,t]$ and $\sum_{(w,a)\in E_2}[s,w]\cdot\mathsf{label}(w,a)\cdot[a,t]$, where $[p,q]$ is the polynomial computed by sub-program in P with source p and sink q. Also, the sets E_1 and E_2 are chosen carefully such that the sub-programs obtained are more or less balanced in terms of the number of variables.

In the following theorem, for nodes u, v in an smABP P, we denote $[u,v]$ is the polynomial computed by sub-program in P with source u and sink v. Also, let $X_{u,v}$ denote the set of all variables that occur as labels in any path from node u to node v in P. More formally, we prove:

▶ **Theorem 1.** *Let P be an smABP of size S computing $f \in \mathbb{F}[x_1,\ldots,x_n]$. There exists edges $\{(u_1,v_1),\ldots,(u_m,v_m)\}$ and $\{(w_1,a_1),\ldots,(w_r,a_r)\}$ in P such that*

(1) *For $i\in[m]$, $n/3 \le |X_{s,u_i}| \le 2n/3$; and*

(2) *For $i\in[r]$, $|X_{s,w_i}| < n/3$ and $|X_{a_i,t}| \le 2n/3$; and*

(3) *$f=\sum_{i=1}^{m}[s,u_i]\cdot\mathit{label}(u_i,v_i)\cdot[v_i,t]+\sum_{i=1}^{r}[s,w_i]\cdot\mathit{label}(w_i,a_i)\cdot[a_i,t]$.*

Let $\Sigma\Pi^{[\sqrt{n}]}(\Sigma\Pi)^{[\sqrt{n}]}$ denote the class of depth four arithmetic circuits where the top layer of Π gates are products of at most $O(\sqrt{n})$ polynomials each being a multilinear polynomial on $O(\sqrt{n})$ variables. As an immediate corollary of the above decomposition, we obtain the following low-arity version of the depth reduction in [1, 22] for the case of smABPs:

▶ **Corollary 1.** *Let P be a syntactic multilinear ABP of size S computing a polynomial f in $\mathbb{F}[x_1, \ldots, x_n]$. Then there exists a $\Sigma\Pi^{[\sqrt{n}]}(\Sigma\Pi)^{[\sqrt{n}]}$ syntactic multilinear formula of size $2^{O(\sqrt{n}\log S)}$ computing f.*

Using the structural property of the parse trees of formulas obtained from smABPs, we prove exponential size lower bounds for two classes of smABPs with restrictions on the variable order.

(1) Strict circular-interval ABPs: A strict circular-interval ABP is an smABP in which the index set of variables in every subprogram is contained in some circular-interval in $\{1, \ldots, n\}$ (see Section 4.1 for a formal definition). Every multilinear polynomial can be computed by a strict circular-interval ABP and hence it is a universal model for computing multilinear polynomials. We obtain an exponential lower bound on the size of any strict circular-interval ABP computing the explicit multilinear polynomial defined by Raz and Yehudayoff in [19] (see Section 2 for definition of the polynomial).

▶ **Theorem 2.** *There exists an explicit multilinear polynomial g in $\mathbb{F}[x_1, \ldots, x_n]$ such that any strict circular-interval ABP computing g requires size $2^{\Omega(\sqrt{n}/\log n)}$.*

Another sub-class of smABPs that we study is the class of *bounded-order* smABPs.

(2) $\mathcal{L}$-ordered smABPs: Jansen [11] introduced *Ordered smABPs* which are branching programs with source s and sink t such that every path from s to t reads variables in a fixed order $\pi \in S_n$. Jansen [11] translated the exponential lower bound for the non-commutative model in [15] to ordered smABPs. Ordered smABPs have also been studied in the context of the polynomial identity testing problem. Further, it is shown in [12] that ordered smABPs are equivalent to Read-Once Oblivious Algebraic Branching Programs (ROABPs for short., see Section 2 for a definition). Polynomial time white-box and quasi-polynomial time black-box algorithms were obtained for identity testing of polynomials computed by ordered smABPs, (see [12, 9, 10] and the references therein).

A natural generalization of ordered smABPs is to allow multiple orders. An smABP is $\mathcal{L}$-ordered if variables can occur along any source to sink path in one of the $\mathcal{L}$ fixed orders. Since there are at most 2^n multilinear monomials in n variables, any multilinear n variate polynomial can be computed by 2^n-ordered ABPs. In this article, by exploiting a simple structural property of $\mathcal{L}$-ordered ABPs, we prove a sub-exponential lower bound for sum of $\mathcal{L}$-ordered smABPs when $\mathcal{L} = O(2^{n^{1/2-\epsilon}})$ for some $\epsilon > 0$:

▶ **Theorem 3.** *Let $\mathcal{L} \leq 2^{n^{1/2-\epsilon}}, \epsilon > 0$ and $f_1, \ldots, f_m \in \mathbb{F}[x_1, \ldots, x_n]$ be multilinear polynomials computed by $\mathcal{L}$-ordered ABPs of size $S_1, \ldots, S_m$ respectively. There exists an explicit multilinear polynomial g in $\mathbb{F}[x_1, \ldots, x_n]$ such that if $g = f_1 + \cdots + f_m$ then either $m = 2^{\Omega(n^{1/40})}$ or there is an $i \in [m]$ such that $S_i = 2^{\Omega(n^{1/40})}$.*

Further, we compare strict circular-interval ABPs and $\mathcal{L}$-ordered ABPs to other mulitlinear models of computation. We show that the construction given in [12] for the equivalence of ROABPs and 1-ordered ABPs can be generalized to $\mathcal{L}$-ordered smABPs. In particular, in Theorem 11 we prove that an $\mathcal{L}$-ordered ABP of size S can be transformed into an equivalent $\mathcal{L}$-pass smABP (see Section 2 for a definition) of size $\mathsf{poly}(S, \mathcal{L})$. Though the overall idea is simple, the construction requires a lot of book-keeping of variable orders. In the context of strict circular-interval ABPs, we compare strict circular-interval ABPs with the simplest class of smABPs: Read-Once Oblivious Algebraic Branching Programs. [15] gives exponential size lower bounds for ROABPs (ABPs on n variables with $n+1$ layers such that every edge from a node in one layer L_i to a node in the successive layer is labeled by a uni-variate polynomial in $\mathbb{F}[x_i]$). Trivially, every strict circular-interval ABP is an ROABP. In Corollary 16, we show that a generalization of strict circular-interval ABPs called *circular-interval* ABPs are more powerful than sum of ROABPs.

Related works. In an independent and simultaneous work, Kumar, Oliveira and Saptharishi [14] show an improved depth reduction (along with several other results) for syntactic multilinear circuits. In [14] (Lemma 6.1 together with Theorem 5.1), the authors show that any *multi-k-ic circuit* of size S can be transformed into an equivalent multi-k-ic $\Sigma\Pi(\Sigma\Pi)^t$ circuit of size $2^{kt} \cdot S^{O(kn/t)}$. Thus, we have $k = 1$ in the case of syntactic multilinear circuits and the circuit size bound in Lemma 6.1 of [14] matches the size bound in Corollary 1 of this paper when $t = \sqrt{n}$. However, Corollary 1 works for syntactic multilinear ABPs while the result in [14] is for the more general class of syntactic multilinear circuits.

In [16], the authors prove lower bounds for sum of $\mathcal{L}$-pass ABPs computing a multilinear polynomial in VP. The result in Theorem 3 is much stronger than the results in [16], since it allows exponentially many orders, whereas the arguments in [16] work only when $\mathcal{L} = o(n^{1/12})$.

Saptharishi and coauthors (through a personal communication) have shown that $\mathcal{L}$-ordered smABPs can be written as a sum of $\mathcal{L}$ ROABPs. While, this will help in significantly improving the lower bound given in Theorem 3, the restriction on $\mathcal{L}$ will be much smaller than $2^{n^{1/2-\epsilon}}$ as far as we are aware.

In [4], Arvind and Raja considered interval ABPs where for every node v reachable from the source, the index set of variables in the sub-program with v as the sink node must be an interval in $[1, n]$. They proved an exponential size lower bound for interval ABPs assuming the *sum-of-squares* conjecture. Although our model is more restrictive than the one in [4], our lower bound argument is unconditional.

Proofs that are omitted due to space constraints can be found int he full version [17].

2 Preliminaries

In this section, we include necessary definitions of models and notations used in this article. Let $\mathbb{F}$ be a field and $X = \{x_1, \ldots, x_n\}$ denote a finite set of variables.

An *arithmetic circuit* $\mathcal{C}$ over $\mathbb{F}$ is a directed acyclic graph with vertices of in-degree zero or two. A vertex of out-degree 0 is called an output gate. A node of in-degree zero is called an input gate and is labeled by elements from $X \cup \mathbb{F}$. Every other gate is labeled either by $+$ or $\times$. Every gate in $\mathcal{C}$ naturally computes a polynomial in $\mathbb{F}[X]$ and the polynomial computed by $\mathcal{C}$ is the polynomial computed at the output gate. When the circuit has more than one output gate, the circuit computes a set of polynomials. The *size* of $\mathcal{C}$ is the number of gates in $\mathcal{C}$ and *depth* of $\mathcal{C}$ is the length of the longest path from an input gate to the output gate in $\mathcal{C}$. An *arithmetic formula* is an arithmetic circuit where the underlying undirected graph is a tree.

A *parse tree* T of an arithmetic formula F is a sub-tree of F containing the output gate of F such that for every $+$ gate v of F that is included in T, exactly one child of v is in T and for every $\times$ gate u that is in T, both children of u are in T.

An *algebraic branching program* (ABP) P is a directed acyclic graph with one vertex s of in-degree zero (source) and one vertex t of out-degree zero (sink). The vertices of the graph are partitioned into layers $L_0, L_1, \ldots, L_\ell$ where edges are from vertices in layer L_i to those in L_{i+1} for every $0 \le i \le \ell - 1$. The source node s is the only vertex in layer L_0 and the sink t is the only vertex in layer L_ℓ. Every edge in e in P is labeled by an element in $X \cup \mathbb{F}$ (denoted by $\mathsf{label}(e)$). The width of P is $\max_i\{|L_i|\}$ and size is the number of nodes in P. For a path ρ the weight, $\mathsf{wt}(\rho)$ be the product of its edge labels. The polynomial computed by an ABP P is the sum of weights of all s to t paths in P. For nodes u and v in P, let $[u, v]_P$ denote the polynomial computed by the sub-program of P with u as the source node and v as the sink node. We drop the subscript from $[u, v]_P$ when P is clear from the context.

Let $X_{u,v}$ denote the set of all variables that occur as labels in any path from u to v in P.

An ABP P is said to be *syntactic multilinear* (smABP) if every variable occurs at most once in every s to t path in P, it is said to be an *oblivious* ABP if for every layer L in P, there is a variable x_{i_L} such that every edge from the layer L is labeled from $\{x_{i_L}\} \cup \mathbb{F}$. A *Read-Once Oblivious* (ROABP) is an oblivious smABP such that every variable appears as an edge label in at most one layer.

Anderson et al. [3] defined the class of $\mathcal{L}$-pass smABPs. An oblivious smABP P is $\mathcal{L}$-pass, if there are layers $i_1 < i_2 < \ldots < i_{\mathcal{L}}$ such that for every j, between layers i_j and i_{j+1} the program P is an ROABP. Let π be a permutation of $\{1, \ldots, n\}$ and P be an smABP computing an n-variate multilinear polynomial. An s to t path ρ in P is said to be *consistent* with π, if the variable labels in ρ occur as per the order given by π, i.e., if x_i and x_j occur as edge labels in ρ in that order, then $\pi(i) < \pi(j)$. For a node v of P, v is said to be consistent with π, if every s to v path is consistent with π.

An smABP P is said to be *$\mathcal{L}$-ordered*, if there are $\mathcal{L}$ permutations $\pi_1, \ldots, \pi_{\mathcal{L}}$ such that for every s to t path ρ in P, there is an $1 \leq i \leq \mathcal{L}$ such that ρ is consistent with π_i.

We now review the partial derivative matrix of a polynomial introduced in [18]. Let $X = Y \cup Z$ be such that $Y \cap Z = \emptyset$ and $|Y| = |Z|$. It is convenient to represent the partition $X = Y \cup Z$ as an injective function $\varphi : X \to Y \cup Z$. For a polynomial f, let f^φ be the polynomial obtained by substituting each variable $x_i = \varphi(x_i)$ for $1 \leq i \leq n$.

▶ **Definition 1** (Partial Derivative Matrix., [18]). *Let $f \in \mathbb{F}[X]$ be a multilinear polynomial. The* partial derivative matrix *of f (denoted by M_f) with respect to the partition $\varphi : X \to Y \cup Z$ is a $2^m \times 2^m$ matrix defined as follows. For multilinear monomials p and q in variables Y and Z respectively, the entry $M_f[p, q]$ is the coefficient of the monomial pq in f^φ.*
For a polynomial f and a partition φ, let $\mathsf{rank}_\varphi(f)$ denote the rank of the matrix M_{f^φ} over the field $\mathbb{F}$. The following properties of $\mathsf{rank}_\varphi(f)$ are useful:

▶ **Lemma 4** ([18], Propositions 3.1, 3.2 and 3.3). *Let $f, g \in \mathbb{F}[Y, Z]$ be multilinear polynomials. Then, (1) $\mathsf{rank}_\varphi(f + g) \leq \mathsf{rank}_\varphi(f) + \mathsf{rank}_\varphi(g)$; (2) If $\mathsf{var}(f) \cap \mathsf{var}(g) = \emptyset$, then $\mathsf{rank}_\varphi(fg) = \mathsf{rank}_\varphi(f) \cdot \mathsf{rank}_\varphi(g)$; and (3) If $f \in \mathbb{F}[Y_1, Z_1]$ for $Y_1 \subseteq Y, Z_1 \subseteq Z$, then $\mathsf{rank}_\varphi(f) \leq 2^{\min\{|Y_1|,|Z_1|\}}$.*

Let $\mathcal{D}$ denote the uniform distribution on the set of all partitions $\varphi : X \to Y \cup Z$, where $|Y| = |Z| = |X|/2$. The following is known about the rank of ROABPs:

▶ **Lemma 5** ([16], Corollary 1). *Let f be an N-variate multilinear polynomial computed by an ROABP of size S. Then,*

$$\Pr_{\varphi \sim \mathcal{D}}[\mathsf{rank}_\varphi(f) \leq S^{\log N} 2^{N/2 - N^{1/5}}] \geq 1 - 2^{-N^{1/5}}.$$

We need the following polynomial defined in [19] to prove lower bounds:

▶ **Definition 2** (Full rank Polynomial., [19]). *Let $n \in \mathbb{N}$ be even and $\mathcal{W} = \{w_{i,k,j}\}_{i,k,j \in [n]}$. For any two integers $i, j \in \mathbb{N}$, we define an interval $[i, j] = \{k \in \mathbb{N}, i \leq k \leq j\}$. Let $|[i, j]| = j - i + 1$, $X_{i,j} = \{x_p \mid p \in [i, j]\}$. Let $\mathbb{G} = \mathbb{F}(\mathcal{W})$, the rational function field. For every $[i, j]$ such that $|[i, j]|$ is even we define a polynomial $g_{i,j} \in \mathbb{G}[X]$ as $g_{i,j} = 1$ when $|[i, j]| = 0$ and if $|[i, j]| > 0$ then, $g_{i,j} \triangleq (1 + x_i x_j) g_{i+1,j-1} + \sum_k w_{i,k,j} g_{i,k} g_{k+1,j}$. where x_k, $w_{i,k,j}$ are distinct variables, $i \leq k \leq j$ and the summation is over $k \in [i + 1, j - 2]$ such that $|[i, k]|$ is even. Let $g \triangleq g_{1,n}$.*

It is known that for any partition $\varphi \sim \mathcal{D}$, $\mathsf{rank}_\varphi(g)$ is the maximum possible value:

▶ **Lemma 6** ([19], Lemma 4.3). *Let $n \in \mathbb{N}$ be even and $\mathbb{G}$ be as above. Let $g \in \mathbb{G}[X]$ be the polynomial in Definition 2. Then for any $\varphi \sim \mathcal{D}$, $\mathsf{rank}_\varphi(g) = 2^{n/2}$.*

3 A variable-balanced decomposition for syntactic multilinear ABPs

In this section, we give a new decomposition for smABPs. The decomposition can be seen as a variable-balanced version of the well known decomposition of arithmetic circuits given by Valiant et al. [24] for the case of smABPs. In fact, we show that a smABP can be divided into sub-programs that are almost balanced in terms of the number of variables.

▶ **Theorem 1.** *Let P be an smABP of size S computing $f \in \mathbb{F}[x_1, \ldots, x_n]$. There exists edges $\{(u_1, v_1), \ldots, (u_m, v_m)\}$ and $\{(w_1, a_1), \ldots, (w_r, a_r)\}$ in P such that*

(1) *For $i \in [m]$, $n/3 \leq |X_{s,u_i}| \leq 2n/3$; and*

(2) *For $i \in [r]$, $|X_{s,w_i}| < n/3$ and $|X_{a_i,t}| \leq 2n/3$; and*

(3) *$f = \sum_{i=1}^{m}[s, u_i] \cdot \textsf{label}(u_i, v_i) \cdot [v_i, t] + \sum_{i=1}^{r}[s, w_i] \cdot \textsf{label}(w_i, a_i) \cdot [a_i, t]$.*

Proof. The proof is by a careful subdivision of the program P. We assume without loss of generality that t is reachable from every node in P and that every node in P has in-degree and out-degree at most 2. Consider the following coloring procedure:

(1) Initialize by coloring t as blue. Repeat (2) until no new node is colored.

(2) Consider node u that is colored blue such that at least one of nodes v or w are uncolored, where (v, u) and (w, u) are the only edges incoming to u. For $a \in \{v, w\}$ do following :

   a. If $|X_{s,a}| > 2n/3$, then color a as blue.

   b. If $n/3 \leq |X_{s,a}| \leq 2n/3$, then color a as red.

   c. If $|X_{s,a}| < n/3$, then color a as green.

At the end of the above coloring procedure we have the following:

1. For every node u with incoming edges (v, u) and (w, u), if u is colored blue then both v and w are colored.
2. For every directed $s \rightsquigarrow t$ path ρ in P, exactly one of the following holds:
   a. ρ has exactly one edge (v, w) such that v is colored red and w is colored blue.
   b. ρ has exactly one edge (v, w) such that v is colored green and w is colored blue.
3. If a node u is colored blue, then every node v reachable from u must have color blue.

Property 1 follows from the fact that a node v is colored if and only if there is an edge (v, u) such that u is colored blue. For property 3, clearly, a node u is colored blue if and only if $|X_{s,u}| > 2n/3$, thus every node reachable from a blue node is also colored blue. For property 2, let ρ be a directed $s \rightsquigarrow t$ path and v be the first node along ρ that is colored blue. Note $|X_{s,s}| = 0$, so s cannot be colored blue. Clearly, every node that follows v in ρ is colored blue and $v \neq s$. Let u be the node that immediately precedes v in ρ, then clearly, u is either red or green. Uniqueness follows from the fact that no node that precedes u in ρ is colored blue and every node that succeeds v in ρ is colored blue, hence there cannot be another such edge.

Let $E_{rb} = \{(u, v) \in P \mid u \text{ is colored red and } v \text{ is colored blue}\}$ and $E_{gb} = \{(u, v) \in P \mid u \text{ is colored green and } v \text{ is colored blue}\}$. Let $E_{rb} = \{(u_1, v_1), \ldots, (u_m, v_m)\}$ and $E_{gb} = \{(w_1, a_1), \ldots, (w_r, a_r)\}$ where $m, r \leq 2S$. We now prove that sets E_{rb} and E_{gb} satisfy the required properties.

(1) For $i \in [m]$, since $(u_i, v_i) \in E_{rb}$, u_i is colored red. By Step 2(b) of the coloring procedure, $n/3 \leq |X_{s,u_i}| \leq 2n/3$.

(2) For $i \in [r]$, since $(w_i, a_i) \in E_{gb}$, w_i is colored green and a_i is colored blue. By Step 2(c) of the coloring procedure, $|X_{s,w_i}| < n/3$ and by Step 2(a), $|X_{s,a_i}| > 2n/3$. Since P is syntactic multilinear, $|X_{s,a_i}| + |X_{a_i,t}| \leq n$ implying $|X_{a_i,t}| \leq n/3$.

(3) By Property 2, $s \rightsquigarrow t$ paths in P are partitioned into paths that have exactly one edge in E_{rb} and paths that have exactly one edge in E_{gb}. Therefore,

$$f = \sum_{\rho: s \rightsquigarrow t} \mathsf{wt}(\rho) = \sum_{\rho: s \rightsquigarrow t,\ \rho \cap E_{rb} \neq \emptyset} \mathsf{wt}(\rho) + \sum_{\rho: s \rightsquigarrow t,\ \rho \cap E_{gb} \neq \emptyset} \mathsf{wt}(\rho)$$
$$= \sum_{i=1}^{m} [s, u_i] \cdot \mathsf{label}(u_i, v_i) \cdot [v_i, t] + \sum_{i=1}^{r} [s, w_i] \cdot \mathsf{label}(w_i, a_i) \cdot [a_i, t]. \quad \blacktriangleleft$$

The above decomposition allows us to obtain low-depth formulas for syntactic multilinear ABPs with quasi-polynomial blow-up in size. In the following, we show that a syntactic multilinear ABP can be computed by a log-depth syntactic multilinear formula where each leaf represents a multilinear polynomial in $O(\sqrt{n})$ variables.

▶ **Lemma 7.** *Let P be a syntactic multilinear ABP of size S computing a multilinear polynomial $f \in \mathbb{F}[x_1, \ldots, x_n]$. Then, there is a syntactic multilinear formula Φ with gates of unbounded fan-in computing f of size $S^{O(\log n)}$ and depth $O(\log n)$ such that every leaf w in Φ represents a multilinear polynomial $[u, v]_{P_w}$ for some nodes u, v in P_w with $|X_{u,v}| \leq \sqrt{n}$, where P_w is a sub-program of P. Further, any parse tree of Φ has at most $3\sqrt{n}$ leaves.*

Proof. Let nodes s and t be source and sink of P. The proof constructs a formula Φ by induction on the number of variables $|X_{s,t}|$ in the program P.
Base Case : If $|X_{s,t}| \leq \sqrt{n}$, then $\phi_{s,t}$ is a leaf gate with label $[s, t]$.
Induction Step : For induction step, suppose $|X_{s,t}| > \sqrt{n}$. By Theorem 1, we have

$$f = \sum_{i=1}^{m} [s, u_i] \cdot \mathsf{label}(u_i, v_i) \cdot [v_i, t] + \sum_{i=1}^{r} [s, w_i] \cdot \mathsf{label}(w_i, a_i) \cdot [a_i, t], \tag{1}$$

where u_i, v_i, w_i and a_i are nodes in P, with $|X_{s,t}|/3 \leq |X_{s,u_i}| \leq 2|X_{s,t}|/3$ and $|X_{s,w_i}| + |X_{a_i,t}| \leq 2|X_{s,t}|/3$. Further, $[s, u_i] \cdot \mathsf{label}(u_i, v_i)$ (resp., $[s, w_i] \cdot \mathsf{label}(w_i, a_i)$) is an smABP with at most $2|X_{s,t}|/3 + 1$ (resp., $|X_{s,t}|/3$) variables. Let

$$f = \sum_{i=1}^{m} g_i h_i + \sum_{i=1}^{r} g'_i h'_i, \tag{2}$$

where $g_i = [s, u_i] \cdot \mathsf{label}(u_i, v_i)$, $h_i = [v_i, t]$, $g'_i = [s, w_i] \cdot \mathsf{label}(w_i, a_i)$ and $h'_i = [a_i, t]$. For any i, if $|X_{s,w_i}| + |X_{a_i,t}| < \sqrt{n}$, then we set $g'_i = [s, w_i] \cdot \mathsf{label}(w_i, a_i) \cdot [a_i, t]$ and $h'_i = 1$. By induction, suppose ϕ_i (resp. ϕ'_i) be the multilinear formula that computes g_i (resp. g'_i) and ψ_i (resp. ψ'_i) be that for h_i (resp. h'_i). Set $\Phi = \sum_{i=1}^{m} (\phi_i \times \psi_i) + \sum_{i=1}^{r} (\phi'_i \times \psi'_i)$. Let $T(n)$ denote the size of the resulting formula on n variables. Then, $T(n) \leq 2 \cdot S \cdot 2 \cdot T(2n/3) = S^{O(\log n)}$. Thus, Φ is a syntactic multilinear formula of size $S^{O(\log n)}$ and depth $O(\log n)$ computing f. By construction, every leaf represents a multilinear polynomial $[u, v]_P$ for some nodes u, v in P with $|X_{u,v}| \leq \sqrt{n}$.

It remains to prove that any parse tree of Φ has at most $3\sqrt{n}$ leaves. We begin with a description of the process for constructing parse sub-trees of Φ. By Equation (2), constructing a parse tree of Φ is equivalent to the process:

1. Choose $b \in \{0, 1\}$ (corresponds to choosing one of the summations in Equation (2)).
2. If $b = 0$ choose $i \in \{1, \ldots, m\}$, else if $b = 1$ choose $j \in \{1, \ldots, r\}$.
3. Repeat steps 1 and 2 for sub-formulas ϕ_i, ϕ'_i, ψ_i and ψ'_i (depending on choice of b).

Consider any parse tree T of Φ. It is enough to prove that every leaf in T that is not labeled by 1 is a polynomial in at least $\sqrt{n}/3$ variables. Since Φ is syntactic multilinear, it follows that any parse tree of Φ has at most $3\sqrt{n}$ leaves, as required. However, it may be noted that this may not be true always. Instead, we argue that every leaf in T can be associated with a set of at least $\sqrt{n}/3$ variables such that no other leaf in T can be associated with these variables, hence implying that the number of leaves in any parse tree T of Φ is at most $3\sqrt{n}$.

Consider a leaf v in T having less than $\sqrt{n}/3$ variables. Let u be the first sum gate on the path from v to root with $|X_u| > \sqrt{n}$. Note that such a node u exists always, excluding the cases when the smABP P is a just a product of variable disjoint ABPs. Rest of the argument is split based on whether $b = 0$ or $b = 1$ at the step for choosing v in the construction of parse tree T. For the remainder of this proof, for any gate u, let X_u denote the set of variables in the sub-formula rooted at u in Φ.

Firstly, suppose that in the construction of T, $b = 0$ at the step for choosing v. Then, either $v = [p, u_i]\cdot\mathsf{label}(u_i, v_i)$ or $v = [v_i, q]$ for some nodes p, q, u_i, v_i in P, where u_i (respectively v_i) is colored red (respectively blue) when the coloring procedure (described in the proof of Theorem 1) is performed on the sub-program with source p and sink q. If $v = [p, u_i] \cdot \mathsf{label}(u_i, v_i)$, $|X_v| \geq |X_{p,u_i}| \geq |X_u|/3 \geq \sqrt{n}/3$, a contradiction to fact that v is a leaf in T with fewer than $\sqrt{n}/3$ variables. Hence, $v = [v_i, q]$. Set $A(v) = X_u \setminus (X_{p,u_i} \cup \{\mathsf{label}(u_i, v_i)\})$. Clearly, as $|X_u| \geq \sqrt{n}$ and $|X_{p,u_i}| \leq 2|X_u|/3$, we have $|A(v)| \geq \sqrt{n}/3$.

When $b = 1$, we have the following possibilities:

Case 1 $v = [p, w_i] \cdot \mathsf{label}(w_i, a_i) \cdot [a_i, q]$. In this case, set $A(v) = X_u$. Then $|A(v)| \geq \sqrt{n}/3$.

Case 2 $v = [p, w_i] \cdot \mathsf{label}(w_i, a_i)$. In this case, set $A(v) = X_u \setminus X_{a_i,q}$. Then $|A(v)| = |X_u| - |X_{a_i,q}| \geq \sqrt{n}/3$ as $|X_{a_i,q}| \leq 2|X_u|/3$ and $|X_u| > \sqrt{n}$.

Case 3 $v = [a_i, q]$. Set $A(v) = X_u \setminus (X_{p,w_i} \cup \{\mathsf{label}(w_i, a_i)\})$. Then $|A(v)| = |X_u| - |X_{p,w_i}| \geq \sqrt{n}/3$ as $|X_{p,w_i}| \leq 2|X_u|/3$ and $|X_u| > \sqrt{n}$.

It remains to prove that, for any two distinct leaves v and v' in T such that $A(v)$ and $A(v')$ are defined, $A(v) \cap A(v') = \emptyset$. Let u and u' respectively be parents of v and v' in T.

When $u = u'$, there are four possibilities for v and v': $v = [p, w_i] \cdot \mathsf{label}(w_i, a_i) \cdot [a_i, q]$ and $v' = 1$, $v = 1$ and $v' = [p, w_i] \cdot \mathsf{label}(w_i, a_i) \cdot [a_i, q]$, $v = [p, w_i] \cdot \mathsf{label}(w_i, a_i)$ and $v' = [a_i, q]$, or $v' = [p, w_i] \cdot \mathsf{label}(w_i, a_i)$ and $v = [a_i, q]$. As $A(v)$ is defined only for non-constant leaves, the only case is when $v = [p, w_i] \cdot \mathsf{label}(w_i, a_i), v' = [a_i, q]$ or vice-versa. In either of the cases, we have $A(v) \cap A(v') = \emptyset$. Now suppose, $u \neq u'$ and $A(v) \cap A(v') \neq \emptyset$. Then, we have $X_u \cap X_{u'} \neq \emptyset$ as $A(v) \subseteq X_u$ and $A(v') \subseteq X_{u'}$. From the fact that u and u' appear in the same parse tree we can conclude that the least common ancestor of u and u' in Φ must be a $\times$ gate. Let $[p, q]$ and $[p'q']$ be the sub-programs of P that correspond to u and u' respectively. By the construction of Φ, we can conclude that either there is a path from q to p' or there is a path from q' to p in P. Either of the cases is a contradiction to the fact that P is syntactic multilinear. ◀

Now, we obtain a reduction to depth-4 formulas for syntactic multilinear ABPs. Denote by $\Sigma^{[T]}\Pi^{[d]}(\Sigma\Pi)^{[r]}$ the class $\Sigma_{i=1}^{T}\Pi_{j=1}^{O(d)}Q_{ij}$ where Q_{ij}'s are mulitlinear polynomials in $O(r)$ variables. As a corollary to Lemma 7, we have the following reduction to syntactic multilinear $\Sigma\Pi^{[\sqrt{n}]}(\Sigma\Pi)^{[\sqrt{n}]}$ formulas for smABPs.

▶ **Corollary 1.** *Let P be a syntactic multilinear ABP of size S computing a polynomial f in $\mathbb{F}[x_1, \ldots, x_n]$. Then there exists a $\Sigma\Pi^{[\sqrt{n}]}(\Sigma\Pi)^{[\sqrt{n}]}$ syntactic multilinear formula of size $2^{O(\sqrt{n}\log S)}$ computing f.*

Proof. Let P be a syntactic multilinear ABP of size S computing a multilinear polynomial f in $\mathbb{F}[x_1, \ldots, x_n]$ and Φ be the equivalent syntactic multilinear formula with the properties mentioned in Lemma 7. The leaves of any parse tree of Φ represents the multilinear polynomial $[u,v]_{P_w}$ for some nodes u,v in P_w, a sub-program of P. As Φ is a formula, every parse tree T of Φ is uniquely identified by the set of $3\sqrt{n}$ leaves in T where every leaf in Φ represents a multilinear polynomial computed by some sub-program $[u,v]_{P_w}$ for some nodes u,v in P_w, a sub-program of P. Hence, it suffices to count the number of sub-programs of P. As every sub-program $([u,v])$ in P is obtained by choosing two vertices u and v in P, there are at most $\binom{S}{2}$ sub-programs of P. The number of parse trees of Φ are $S^{O(\sqrt{n})} = 2^{O(\sqrt{n}\log S)}$. As $f = \sum_{T:\text{parse tree of }\Phi} m(T)$ where $m(T)$ is the product of multilinear polynomials corresponding to the leaves of Φ in T, there exists a $\Sigma\Pi^{[\sqrt{n}]}(\Sigma\Pi)^{[\sqrt{n}]}$ syntactic multilinear formula with gates of unbounded fan-in of size $2^{O(\sqrt{n}\log S)}$ computing f. ◀

4 Lower Bounds for special classes of smABPs

This section is devoted to lower bounds for restricted classes of smABPs. Our arguments rely on the depth reduction proved in Section 3 and the full rank polynomial given by Raz and Yehhudayoff [19] (see Defintion 2).

4.1 Lower Bounds for strict circular-interval ABPs

In this section, we prove an exponential size lower bound for a special class of smABPs that we call as strict circular-interval ABPs.

An set $I \subseteq \{1,\ldots,n\}$ is a circular π-interval if $I = \{\pi(i),\pi(i+1),\ldots,\pi(j)\}$ for some $i,j \in [n], i<j$ or $I = \{\pi(i),\pi(i+1),\ldots,\pi(n),\pi(1),\ldots,\pi(j)\}$ for some $i,j\in[n], i>j$. These intervals are called *circular intervals* as every such interval $[i,j]$ in $\{1,\ldots,n\}$ can be viewed as a chord on the circle containing n points. Two circular intervals I and J are said to be *overlapping* if the corresponding chords in the circle intersect and *non-overlapping* otherwise. We define a special class of syntactic multilinear ABPs where the set of variables involved in every sub-program is in some circular π-interval.

▶ **Definition 3** (Strict circular-interval ABP). *Let $\pi \in S_n$ be a permutation. A syntactic mulitlinear ABP P is said to be a* strict π-circular-interval ABP *if*

1. *For any pair of nodes u,v in P, the index set of $X_{u,v}$ is contained in some circular π-interval I_{uv} in $[1,n]$; and*
2. *For any u,a,v in P, the circular π-intervals I_{ua} and I_{av} are non-overlapping.*

P is said to be strict circular-interval ABP if it is a strict π-circular-interval ABP for some permutation π.

We require a few preliminaries to prove the lower bound:

1. For every permutation π in S_n, define the partition $\varphi_\pi : X \to Y \cup Z$ such that
$$\text{for all } 1 \le i \le n/2 \;\; \varphi(x_{\pi(i)}) = y_i \text{ and } \varphi(x_{\pi(n/2+i)}) = z_i. \tag{3}$$
2. For any π in S_n, $|\varphi_\pi(X)\cap Y| = |\varphi_\pi(X)\cap Z| = |X|/2$. For the polynomial g in Definition 2, $\mathsf{rank}_{\varphi_\pi}(g) = 2^{n/2}$ by Lemma 6.
3. For any set $X_i \subseteq X$, let $\varphi_\pi(X_i) = \{\varphi_\pi(x) \mid x \in X_i\}$. We say X_i is *monochromatic* if either $\varphi_\pi(X_i)\cap Y = \emptyset$ or $\varphi_\pi(X_i)\cap Z = \emptyset$. Observe that if X_i is monochromatic then for any polynomial $p_i \in \mathbb{F}[X_i]$, we have $\mathsf{rank}_{\varphi_\pi}(p_i) \le 1$. Further, we say set $X_i \subseteq X$ is *bi-chromatic* if $\varphi_\pi(X_i)\cap Y \ne \emptyset$ and $\varphi_\pi(X_i)\cap Z \ne \emptyset$.

In the following theorem, we show that for any strict circular-interval ABP P computing a polynomial f, there is a partition φ such that $\mathsf{rank}_\varphi(f)$ is small.

▶ **Theorem 8.** *Let P be a strict circular-interval ABP of size S computing f in $\mathbb{F}[x_1,\ldots,x_n]$. There exists a $\varphi : X \to Y \cup Z$ with $|\varphi(X)\cap Y| = |\varphi(X)\cap Z| = |X|/2$ such that $\mathsf{rank}_{\varphi}(f) \le 2^{\sqrt{n}\log n\log S+\sqrt{n}}$.*

Proof. Let Φ be the syntactic multilinear formula constructed from P as given by Lemma 7. Note that any parse tree of Φ has at most $3\sqrt{n}$ leaves. The number of parse trees of Φ is at most $\binom{2^{O(\sqrt{n}\log n\log S)}}{3\sqrt{n}} \le 2^{\epsilon\sqrt{n}\log n\log S}$. Let T be any parse tree of Φ with leaves $w_1,\ldots,w_\ell$ computing polynomials $p_1,\ldots,p_\ell$. We have $f = \sum_{T:\text{parse tree of }\Phi} m(T)$ where $m(T)$ be the product of multilinear polynomials corresponding to the leaves of Φ in T. Let $X_1,\ldots,X_\ell \subseteq X$ be such that p_i is a polynomial in $\mathbb{F}[X_i]$. For every $i\in[\ell]$, let $M_i = \{j \mid x_j \in X_i\}$ be the index set of X_i. As P is a strict circular-interval ABP, we have that sets $M_1,\ldots,M_\ell$ are circular π-intervals in $\{1,\ldots,n\}$ for some $\pi\in S_n$. Let $\varphi_\pi : X \to Y\cup Z$ be the partition function described in Equation (3). If X_i is bi-chromatic then $\mathsf{rank}_{\varphi_\pi}(p_i) \le 2^{\sqrt{n}/2}$ as $|X_i| \le \sqrt{n}$ by construction of formula Φ when w_i is a leaf in Φ.

A crucial observation is that for any parse tree T of Φ, at most two of $\varphi_\pi(X_1),\ldots,\varphi_\pi(X_\ell)$ are bi-chromatic. This is because the existence of bi-chromatic sets $\varphi_\pi(X_i),\varphi_\pi(X_j),\varphi_\pi(X_k)$ for some $i,j,k\in[\ell]$ implies that the circular π-intervals M_i,M_j,M_k are overlapping from the way partition φ_π is defined. As X_i,X_j,X_k are variable sets associated with leaves of the same parse tree T, we can conclude that when $\varphi_\pi(X_i),\varphi_\pi(X_j),\varphi_\pi(X_k)$ are bi-chromatic there exists nodes u,a,v in P such that circular π-intervals I_{ua} and I_{av} are overlapping, a contradiction to the fact that P is a strict circular-interval ABP.

Therefore, in any parse tree T of Φ, at most two of $\varphi_\pi(X_1),\ldots,\varphi_\pi(X_\ell)$ are bi-chromatic say $\varphi_\pi(X_i)$ and $\varphi_\pi(X_j)$. Hence $\mathsf{rank}_{\varphi_\pi}(p_i) \le 2^{\sqrt{n}/2}$ and $\mathsf{rank}_{\varphi_\pi}(p_j) \le 2^{\sqrt{n}/2}$. Also, $\mathsf{rank}_{\varphi_\pi}(p_k) \le 1$ for all $k \ne i,j$. Thus, $\mathsf{rank}_{\varphi_\pi}(f) \le 2^{\epsilon\log n\log S\sqrt{n}+\sqrt{n}}$. ◀

With the above, we can prove Theorem 2:

▶ **Theorem 2.** *There exists an explicit multilinear polynomial g in $\mathbb{F}[x_1,\ldots,x_n]$ such that any strict circular-interval ABP computing g requires size $2^{\Omega(\sqrt{n}/\log n)}$.*

Proof. Let P be a strict circular-interval ABP of size $S = 2^{o(\sqrt{n}/\log n)}$ computing g and Φ be the syntactic multilinear formula obtained from P using Lemma 7. By Theorem 8, there exists a partition $\varphi : X \to Y\cup Z$ with $|\varphi(X)\cap Y| = |\varphi(X)\cap Z| = |X|/2$ such that $\mathsf{rank}_{\varphi}(g) \le 2^{\sqrt{n}+\epsilon\log n\log S\sqrt{n}} < 2^{n/2}$. However, by Lemma 6, $\mathsf{rank}_{\varphi}(g) = 2^{n/2}$, a contradiction. Hence, $S = 2^{\Omega(\sqrt{n}/\log n)}$. ◀

4.2 Lower bound for sum of $\mathcal{L}$-ordered ABPs

In this section, we show that by observing a simple property of the ABP to formula conversion given in Lemma 7, we can obtain lower bounds for $\mathcal{L}$-ordered ABPs for larger sub-exponential values $\mathcal{L}$. In the following lemma, we observe that in the formula obtained from an $\mathcal{L}$-ordered ABP using Lemma 7, a lot of the leaves in any parse tree are in fact 1-ordered ABPs:

▶ **Lemma 9.** *Let P be an $\mathcal{L}$-ordered ABP and F be the syntactic multilinear formula obtained from P using Lemma 7. Then, for any parse tree T of F, all but at most $O(\log\mathcal{L})$ many leaves of T are 1-ordered ABPs (ROABPs).*

Proof. Let T be any parse tree of F with leaves $w_1,\ldots,w_\ell$ and let $p_1,\ldots,p_\ell$ be the polynomials labeling $w_1,\ldots,w_\ell$. From the construction given in the proof of Lemma 7, corresponding to each leaf w_i there are nodes u_i,v_i in P such that polynomial $p_i = [u_i,v_i]\cdot\mathsf{label}(v_i,u_{i+1})$. Consider the syntactic multilinear ABP P' obtained by placing programs

$$[u_1,v_1]\cdot\mathsf{label}(v_1,u_2),[u_2,v_2]\cdot\mathsf{label}(v_2,u_3),\ldots,[u_i,v_i]\cdot\mathsf{label}(v_i,u_{i+1}),\ldots,[u_\ell,v_\ell]$$

in the above order. From the construction above, P' is a sub-program of P and hence the number of variable orders in P' is a lower bound on the number of variable orders in P. If r_i is the number of variable orders in the sub-program $[u_i, v_i]$, the total number of variable orders in the sub-program P' (and hence P) is at least $r_1 \cdot r_2 \cdots r_\ell$. Since the number of distinct orders is at most $\mathcal{L}$, we conclude that $|\{i \mid r_i \geq 2\}| \leq \log \mathcal{L}$, as required. ◀

Let $\mathcal{D}$ denote the uniform distribution on the set of all partitions $\varphi : X \to Y \cup Z$ with $|Y| = |Z|$. In the following lemma, we show that rank of a polynomial computed by an $\mathcal{L}$-ordered ABP is far from being full. Proof can be found in the full version of the article [17].

▶ **Lemma 10.** *Let P be an $\mathcal{L}$-ordered ABP of size S computing a polynomial f in $\mathbb{F}[x_1, \ldots, x_n]$. Then for $k = n^{1/20}$, $\Pr_{\varphi \sim \mathcal{D}}[\mathsf{rank}_\varphi(f) > 2^{\log n \log S \sqrt{n}} \cdot 2^{n/2 - k\sqrt{n}}] \leq S^2 \cdot 2^{-O(n^{1/20})}$.*

Now, we are ready to prove Theorem 3:

▶ **Theorem 3.** *Let $\mathcal{L} \leq 2^{n^{1/2-\epsilon}}, \epsilon > 0$ and $f_1, \ldots, f_m \in \mathbb{F}[x_1, \ldots, x_n]$ be multilinear polynomials computed by $\mathcal{L}$-ordered ABPs of size $S_1, \ldots, S_m$ respectively. There exists an explicit multilinear polynomial g in $\mathbb{F}[x_1, \ldots, x_n]$ such that if $g = f_1 + \cdots + f_m$ then either $m = 2^{\Omega(n^{1/40})}$ or there is an $i \in [m]$ such that $S_i = 2^{\Omega(n^{1/40})}$.*

Proof. Set $k = n^{1/20}$. Suppose, for every i, f_i is computed by $\mathcal{L}$-ordered ABP of size $2^{n^{1/40}}$. Then, $\mathsf{rank}_\varphi(f_i) > 2^{\sqrt{n} \log n \log(2^{n^{1/40}})} 2^{n/2 - k\sqrt{n}}$ with probability at most $2^{2n^{1/40}} 2^{-n^{1/20}}$ when $\varphi \sim \mathcal{D}$. Therefore, probability that there is a i such that $\mathsf{rank}_\varphi(f_i) > 2^{\sqrt{n} \log n \log S} 2^{n/2 - k\sqrt{n}}$ is at most $m 2^{2n^{1/40}} 2^{-n^{1/20}} < 1$ for $m < 2^{n^{1/40}}$. By union bound, there is a $\varphi \sim \mathcal{D}$ such that for every i, $\mathsf{rank}_\varphi(f_i) < 2^{\sqrt{n} \log n \log(2^{n^{1/40}})} 2^{n/2 - k\sqrt{n}} < 2^{n/2}$. But by Lemma 6, $\mathsf{rank}_\varphi(g) = 2^{n/2}$ for every partition φ, which is a contradiction. Hence, either $m = 2^{\Omega(n^{1/40})}$ or there is an $i \in [m]$ such that $S_i = 2^{\Omega(n^{1/40})}$. ◀

5 Comparison with other multilinear circuit models

In this section, we compare strict circular-interval ABPs and $\mathcal{L}$-ordered ABPs to other well known models for computing mulitlinear polynomials.

5.1 $\mathcal{L}$-ordered to $\mathcal{L}$-pass

In this section, we show that $\mathcal{L}$-ordered ABPs can be transformed into ABPs that make at most $\mathcal{L}$ passes on the input, although in different orders.

▶ **Theorem 11.** *Let P be an $\mathcal{L}$-ordered ABP of size S computing a polynomial $f \in \mathbb{F}[x_1, \ldots, x_n]$. Then there is an $\mathcal{L}$-pass ABP Q of size $\mathsf{poly}(\mathcal{L}, S)$ computing f.*

Proof. Let P be an $\mathcal{L}$-ordered ABP of size S computing a polynomial f. Let $L_0, L_1, \ldots, L_\ell$ be the layers of P where source s and sink t are the only nodes in layers L_0 and L_ℓ respectively. Let $u_{i1}, \ldots, u_{iw}$ be nodes in L_i, where $w \leq S$ is the width of P. Without loss of generality assume every node in P has in-degree and out-degree at most two, and every layer except L_0 and L_ℓ has exactly w nodes. Also, every s to t path in P respects one of the permutations $\pi_1, \pi_2, \ldots, \pi_{\mathcal{L}}$. We now construct an $\mathcal{L}$-pass ABP $\mathcal{Q}$ that reads variables in the order $(x_{\pi_1(1)}, x_{\pi_1(2)}, \ldots, x_{\pi_1(n)}), \ldots, (x_{\pi_{\mathcal{L}}(1)}, x_{\pi_{\mathcal{L}}(2)}, \ldots, x_{\pi_{\mathcal{L}}(n)})$. The source and sink of ABP $\mathcal{Q}$ are denoted by s' and t' respectively. The number of layers in $\mathcal{Q}$ will be bounded by $\mathcal{L}(\ell + 1)$ and are labeled as $L_{ir}, i \in [\mathcal{L}], r \in \{0, \ldots, \ell\}$. Intuitively, for a node u_{rj} in layer L_r in P, we have $\mathcal{L}$ copies, $u_{1rj}, u_{2rj} \ldots, u_{\mathcal{L}rj}$ in $\mathcal{Q}$, where u_{irj} is a vertex in layer L_{ir}.

Intuitively, u_{irj} would have all paths from s to u_{rj} that respect the permutation π_i, but none of the permutations π_p for $p < i$. To ensure that the resulting ABP is $\mathcal{L}$-pass, we place the layers as follows : $L_{11}, \ldots, L_{1\ell}, L_{21}, \ldots, L_{2\ell}, \ldots, L_{\mathcal{L}1}, \ldots, L_{\mathcal{L}\ell}$. We construct $\mathcal{Q}$ as follows :

(1) Base Case : In ABP P, for every edge e from source s in layer L_0 to node $u_{1j}, j \leq w$ in layer L_1 labeled by $\mathsf{label}(e) \in X \cup \mathbb{F}$, if $\mathsf{label}(e) = x_k$, then add the edge (s', u_{m1j}) with label x_k where m is the smallest value such that x_k is consistent with π_m, if $\mathsf{label}(e) = \alpha \in \mathbb{F}$, then add the edge (s', u_{m1j}) with label α.

(2) Induction Step : Consider layer $L_r, r \in \{1, \ldots, \ell\}$:

a. For every node u_{rj} in layer L_r of P, with $1 \leq j \leq w$ and every edge e of the form $e = (u_{rj}, u_{r+1,j'})$ do the following:

Case 1: $\mathsf{label}(e) = x_k \in X$. For every $1 \leq i \leq \mathcal{L}$, let m be the smallest index such that every path from s' to u_{irj} concatenated with the edge e is consistent with π_m. Note that, by the construction, $m \geq i$. Add the edge $(u_{irj}, u_{mr+1j'})$ in $\mathcal{Q}$ for every i with label x_k. For every $1 \leq i \leq \mathcal{L}$, note that the choice of m is unique.

Case 2: $\mathsf{label}(e) = \alpha \in \mathbb{F}$. For every $1 \leq i \leq \mathcal{L}$, add edge (u_{irj}, u_{ir+1j}) with label α.

b. Create the node t' in $\mathcal{Q}$, and add edges $(u_{i\ell 1}, t')$ with label 1 for every $1 \leq i \leq \mathcal{L}$.

Note that in the above construction, the resulting branching program will not be layered. It can be made layered by adding suitable new vertices and edges labeled by $1 \in \mathbb{F}$. The correctness of the construction follows from the following claim whose proof is can be found in the full version [17].

▷ Claim 12.

(1) $\mathcal{Q}$ is an $\mathcal{L}$-pass syntactic multilinear ABP and has size $\mathsf{poly}(\mathcal{L}, S)$.

(2) For $1 \leq r \leq \ell$ and node u_{rj} in layer L_r in P, $1 \leq j \leq w$, $[s, u_{rj}]_P = \sum_{i=1}^{\mathcal{L}} [s', u_{irj}]_{\mathcal{Q}}$. ◀

5.2 Circular-Interval ABP vs. Sum of ROABPs

In this section, we define *circular-interval* ABPs (a generalization of strict circular-interval ABPs) and compare them to sum of ROABPs (and sum of strict circular-interval ABPs).

▶ **Definition 4** (Circular-Interval ABP). *Let $\pi \in S_n$ be a permutation. A syntactic mulitlinear ABP P is said to a* π-circular-interval ABP *if for any node v in P, the index set of $X_{s,v}$ is contained in some circular π-interval I_{sv} in $[1, n]$. P is said to be* circular-interval ABP *if it is a π-circular-interval ABP for some permutation π in S_n.*

Let $h = (h_n)_{n \geq 0}$ be the family of multilinear polynomials defined by Dvir et al. [8].

The following properties of the polynomial h are straightforward from Theorem 3.4 of [8] and the definition of circular-interval ABPs:

▶ **Lemma 13** ([8], Theorem 3.4). *(i) Over any field $\mathbb{F}$, the mulitlinear ABP R computing h is a circular-interval ABP of polynomial size. (ii) For any partition $\Pi \sim \mathcal{D}$, $\mathsf{rank}_\Pi(h) = 2^{n/2}$.*

Now, in order to separate circular-interval ABPs from ROABPs, it suffices to construct one partition Π such that $\mathsf{rank}_\Pi(f) < 2^{n/2}$ where f is the polynomial computed by an ROABP. This is guaranteed by the following lemma, whose proof is based on the ideas in [8] and [16]:

▶ **Lemma 14.** *Let Q be an ROABP computing a multilinear polynomial $f \in \mathbb{F}[x_1, \ldots, x_N]$ and Φ_Q be the multilinear formula obtained from Q computing f. Then $\mathsf{rank}_\Pi(f) \leq |\Phi_Q| \cdot 2^{n/2 - n^{1/5000}}$ with probability at least $1 - n^{\Omega(\log n)}$ for $\Pi \sim \mathcal{D}$.*

▶ **Theorem 15.** *Let $f_1, \dots f_m$ in $\mathbb{F}[x_1, \dots, x_n]$ be multilinear polynomials computed by ROABPs of size $R_1, \dots, R_m$ respectively. There exists an explicit multilinear polynomial h in $\mathbb{F}[x_1, \dots, x_n]$ computable by circular-interval ABPs of polynomial size, such that if $h = f_1 + \dots + f_m$ then, either $m = n^{\Omega(\log n)}$ or there is an $i \in [m]$ with $R_i = 2^{\Omega\left(n^{1/6000}/\log N\right)}$.*

As an immediate corollary to Theorem 15, we have the following:

▶ **Corollary 16.** *There is a super-polynomial separation between ROABPs and circular interval ABPs.*

References

1 Manindra Agrawal and V. Vinay. Arithmetic Circuits: A Chasm at Depth Four. In *FOCS*, pages 67–75, 2008. doi:10.1109/FOCS.

2 Noga Alon, Mrinal Kumar, and Ben Lee Volk. Unbalancing Sets and an Almost Quadratic Lower Bound for Syntactically Multilinear Arithmetic Circuits. In *33rd Computational Complexity Conference, CCC 2018, June 22-24, 2018, San Diego, CA, USA*, pages 11:1–11:16, 2018. doi:10.4230/LIPIcs.CCC.2018.11.

3 Matthew Anderson, Michael A. Forbes, Ramprasad Saptharishi, Amir Shpilka, and Ben Lee Volk. Identity Testing and Lower Bounds for Read-k Oblivious Algebraic Branching Programs. *TOCT*, 10(1):3:1–3:30, 2018. doi:10.1145/3170709.

4 Vikraman Arvind and S. Raja. Some Lower Bound Results for Set-Multilinear Arithmetic Computations. *Chicago J. Theor. Comput. Sci.*, 2016, 2016. URL: http://cjtcs.cs.uchicago.edu/articles/2016/6/contents.html.

5 Walter Baur and Volker Strassen. The Complexity of Partial Derivatives. *Theor. Comput. Sci.*, 22:317–330, 1983. doi:10.1016/0304-3975(83)90110-X.

6 Suryajith Chillara, Christian Engels, Nutan Limaye, and Srikanth Srinivasan. A Near-Optimal Depth-Hierarchy Theorem for Small-Depth Multilinear Circuits. *Electronic Colloquium on Computational Complexity (ECCC)*, 25:62, 2018. URL: https://eccc.weizmann.ac.il/report/2018/062.

7 Suryajith Chillara, Nutan Limaye, and Srikanth Srinivasan. Small-depth Multilinear Formula Lower Bounds for Iterated Matrix Multiplication, with Applications. In *35th Symposium on Theoretical Aspects of Computer Science, STACS 2018, February 28 to March 3, 2018, Caen, France*, pages 21:1–21:15, 2018. doi:10.4230/LIPIcs.STACS.2018.21.

8 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.

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. doi:10.1145/2591796.2591816.

10 Rohit Gurjar, Arpita Korwar, and Nitin Saxena. Identity Testing for Constant-Width, and Any-Order, Read-Once Oblivious Arithmetic Branching Programs. *Theory of Computing*, 13(1):1–21, 2017. doi:10.4086/toc.2017.v013a002.

11 Maurice J. Jansen. Lower Bounds for Syntactically Multilinear Algebraic Branching Programs. In *Mathematical Foundations of Computer Science 2008, 33rd International Symposium, MFCS 2008, Torun, Poland, August 25-29, 2008, Proceedings*, pages 407–418, 2008. doi:10.1007/978-3-540-85238-4_33.

12 Maurice J. Jansen, Youming Qiao, and Jayalal Sarma. Deterministic Black-Box Identity Testing π-Ordered Algebraic Branching Programs. In *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2010, December 15-18, 2010, Chennai, India*, pages 296–307, 2010. doi:10.4230/LIPIcs.FSTTCS.2010.296.

13 Mrinal Kumar. A Quadratic Lower Bound for Homogeneous Algebraic Branching Programs. In Ryan O'Donnell, editor, *32nd Computational Complexity Conference (CCC 2017)*, volume 79 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 19:1–19:16. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2017. `doi:10.4230/LIPIcs.CCC.2017.19`.

14 Mrinal Kumar, Rafael Mendes de Oliveira, and Ramprasad Saptharishi. Towards Optimal Depth Reductions for Syntactically Multilinear Circuits. *Electronic Colloquium on Computational Complexity (ECCC)*, 26:19, 2019. URL: `https://eccc.weizmann.ac.il/report/2019/019`.

15 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`.

16 C. Ramya and B. V. Raghavendra Rao. Lower Bounds for Special Cases of Syntactic Multilinear ABPs. In *Computing and Combinatorics - 24th International Conference, COCOON 2018, Qing Dao, China, July 2-4, 2018, Proceedings*, pages 701–712, 2018. `doi:10.1007/978-3-319-94776-1_58`.

17 C. Ramya and B. V. Raghavendra Rao. Lower bounds for multilinear bounded order ABPs. *CoRR*, abs/1901.04377, 2019. `arXiv:1901.04377`.

18 Ran Raz. Multi-linear formulas for permanent and determinant are of super-polynomial size. *J. ACM*, 56(2), 2009. `doi:10.1145/1502793.1502797`.

19 Ran Raz and Amir Yehudayoff. Balancing Syntactically Multilinear Arithmetic Circuits. *Computational Complexity*, 17(4):515–535, 2008. `doi:10.1007/s00037-008-0254-0`.

20 Ran Raz and Amir Yehudayoff. Lower Bounds and Separations for Constant Depth Multilinear Circuits. *Computational Complexity*, 18(2):171–207, 2009. `doi:10.1007/s00037-009-0270-8`.

21 Ramprasad Saptharishi. A survey of lower bounds in arithmetic circuit complexity, 2017. URL: `https://github.com/dasarpmar/lowerbounds-survey/releases`.

22 Sébastien Tavenas. Improved Bounds for Reduction to Depth 4 and Depth 3. In *MFCS*, pages 813–824, 2013. `doi:10.1007/978-3-642-40313-2_71`.

23 Leslie G. Valiant. Completeness Classes in Algebra. In *STOC*, pages 249–261, 1979. `doi:10.1145/800135.804419`.

24 Leslie G. Valiant, Sven Skyum, S. Berkowitz, and Charles Rackoff. Fast Parallel Computation of Polynomials Using Few Processors. *SIAM J. Comput.*, 12(4):641–644, 1983. `doi:10.1137/0212043`.