Lower Bounds for Constant Depth Arithmetic Circuits: The LST-Bound

Notes by Thomas Thierauf

December 22, 2022

Nutan Limaye, Srikanth Srinivasan and Sébastien Tavenas [LST21] proved the first super-polynomial lower bounds for constant depth arithmetic circuits. They got the FOCS 2021 best paper award for it.

1 Introduction

How many arithmetic operations are needed to compute a polynomial? Let us look at an example. Define

$$p(x_1, x_2, \ldots, x_n) = \sum_{S \subseteq [n]} \prod_{i \in S} x_i \,. \tag{1}$$

If one implements p as a circuit according to (1), the Σ-gate has fan-in 2^n. So the whole circuit has size $O(n2^n)$. However, there is a much smaller circuit for p because we have

$$p(x_1, x_2, \ldots, x_n) = \prod_{i \in [n]} (x_i + 1) \,.$$

Hence, p has a circuit of size $O(n)$.

A major challenge is to prove lower bounds for the circuit size of explicit polynomials. Before the LST-result only polynomial lower bounds were known. The best bounds were

- $\Omega(n^3/\log^2 n)$, for $\Sigma\Pi\Sigma$-circuits, by Kayal, Saha, Tavenas (2016),
- $\Omega(n^{2.5})$, for $\Sigma\Pi\Sigma\Pi$-circuits, by Gupta, Saha, Thankey (2020).

The new lower bound for $\Sigma\Pi\Sigma$-circuits is a super-polynomial bound: $n^{\Omega(\sqrt{d})}$, for a polynomial of degree d. Hence, this is a major step ahead in proving lower bounds!

2 Preliminaries

Let $\mathbb{F}$ be a field of characteristic zero and $f \in \mathbb{F}[X]$ be a multivariate polynomial of (total) degree d over a set X of variables. Polynomial f is *homogeneous*, if every nonzero monomial of f has exactly degree d. We call f *multilinear*, if every variable of f has individual degree 1. Let $(X_1, X_2, \ldots, X_d)$ be a partition of the variable set X. We say that f is *set-multilinear* with respect to this partition,

if f is a homogeneous multilinear polynomial of degree d such that every nonzero monomial of f has exactly one variable from every X_i, for $i = 1, 2, \ldots, d$.

Let C be an arithmetic circuit that computes some homogeneous polynomial. We call circuit C *homogeneous*, if every gate of C computes an homogeneous polynomial. Similarly, we say that C is *set-multilinear*, if every gate of C computes a set-multilinear polynomial. In more detail, let C compute a set-multilinear polynomial for a partition $(X_1, X_2, \ldots, X_d)$ of the variables. The gates of C then compute set-multilinear polynomials over sub-partitions $(X_{i_1}, X_{i_2}, \ldots, X_{i_{d'}})$, for some $d' \leq d$. For a $+$-gate of C, the two input polynomial must be over the same sub-partition, whereas for a $\times$-gate, they must be over disjoint sub-partitions.

3 Theorem

The lower bounds are proved for the *Iterated Matrix Multiplication (IMM)* problem which is defined as follows. Given $m \times m$ matrices $X_1, X_2, \ldots, X_d$, for some $m, d \geq 1$. The task is to compute the $(1,1)$-entry of the product $X_1 X_2 \cdots X_d$,

$$\mathrm{IMM}_{m,d}(X_1, X_2, \ldots, X_d) = (X_1 X_2 \cdots X_d)_{1,1}\,.$$

Theorem 3.1 ([LST21]). *Let d, m such that $d = o(\log m)$. Let $\Delta > 0$ be some constant. Any circuit of product-depth Δ that computes $\mathrm{IMM}_{m,d}$ over $\mathbb{F}$ has size $m^{d^{\frac{1}{\exp(\Delta)}}}$.*

Here, we will focus on the special case of Theorem 3.1 where $\Delta = 1$.

Theorem 3.2 ([LST21]). *Let d, m such that $d = O(\log m)$. Any $\Sigma\Pi\Sigma$-circuit that computes $\mathrm{IMM}_{m,d}$ over $\mathbb{F}$ has size $m^{\Omega(\sqrt{d})}$.*

The lower bound for $\Sigma\Pi\Sigma$-circuits is tight, up to constants in the exponent. In this note, we present a proof of the simplest case, i.e. of Theorem 3.2. We try to keep the proof as simple as possible, and even over-simplify a bit by ignoring some issues in the choice of the parameters.

The argument is split into the following three sections. In Section 4, we show how to transform a given general circuit of depth 3 into an equivalent set-multilinear circuit of depth 5 with slightly larger size. A reader who just wants to understand the lower bound for set-multilinear circuits can safely skip this section.

In Section 5, we define the complexity measure and give some examples of its applications. Then, in Section 6, we define the hard polynomial, a projection of IMM, and show that set-multilinear circuits of depth 5 that compute it have super-polynomial size. Reversing the above transformations will prove Theorem 3.2.

4 Transformation into a set-multilinear circuit

Let f be a set-multilinear polynomial with n variables and degree d, and let C be a circuit that computes f with product-depth Δ and size s. We first transform C into an equivalent *set-multilinear circuit* C'. The transformation is done in two steps by the following two lemmas.

Lemma 4.1 (Homogenization). *Let* C *be a circuit of size* s *and product-depth* Δ *that computes an* n*-variate homogeneous polynomial* f *of degree* d*. Then there is a circuit equivalent to* C *that is* homogeneous*, of product-depth* 2Δ *and size* $2^{O(\sqrt{d})}\,\mathrm{poly}(s)$*.*

Proof. We show the case $\Delta = 1$. That is, given a $\Sigma\Pi\Sigma$-circuit C, we show that there exists an equivalent *homogeneous* $\Sigma\Pi\Sigma\Pi\Sigma$-circuit of size $2^{O(\sqrt{d})}\,\mathrm{poly}(s)$.

Let q be the polynomial computed at a product-gate of C. Let $q = \prod_{i=1}^{s} \ell_i$, where the ℓ_i's are linear polynomials. Our goal is to construct a homogeneous circuit that computes $q^{(d)}$, the homogeneous degree-d part of q. Then we are done because it just remains to sum up these $q^{(d)}$'s to compute f. Note that the terms of degree different from d must cancel themselves, as the outcome of C, i.e. f, has no such terms. These cancellations we now implement in the circuitry by ignoring them.

Suppose all the ℓ_i's have *no* constant term. Then the product gate that computes q is already homogeneous. If $s = d$, then we have $q = q^{(d)}$ and we are done.

Inhomogeneity comes with the constant terms. Hence, the interesting case is when only, say, $d' < d$ of the ℓ_i's have constant term 0. These we can leave unchanged. Then it suffices to extract the homogeneous degree-$(d - d')$ part from the remaining $s - d'$ polynomials ℓ_i. For simplicity of notation, we assume now that *all* the ℓ_i's have a nonzero constant term. Moreover, we can extract the constant terms as factors from each ℓ_i, so that all constant terms become 1. Hence, it suffices to consider the case where $q = c\prod_{i=1}^{s} \ell_i$, for some constant c, and

$$\ell_i = \ell_i' + 1,$$

where ℓ_i', is the homogeneous degree-1 part of ℓ_i, for $i = 1, 2, \ldots, s$.

There is an obvious way to express $q^{(d)}$ now:

$$q^{(d)} = c \sum_{\substack{T \subseteq [s] \\ |T| = d}} \prod_{i \in T} \ell_i'. \tag{2}$$

When we use (2) directly to replace the product-gate in C, we can merge the outer sum-gate in (2) with the output sum-gate from C and get a circuit of depth 3, same as C. However, the size would become $\geq \binom{s}{d}$, i.e. around s^d which is too much for our purpose.

Note that (2) says that $q^{(d)}$ essentially is the *elementary symmetric polynomial* $S_{s,d}$,

$$q^{(d)} = c\, S_{s,d}(\ell_1', \ldots, \ell_s').$$

The *Newton Identities* express elementary symmetric polynomials in terms of *power sums* $P_{s,k}$,

$$P_{s,k}(x_1, \ldots x_s) = \sum_{i=1}^{s} x_i^k.$$

Newton Identities are usually stated in the following form:

Lemma 4.2 (Newton Identities). *For* $k \in [s]$*,*

$$k\, S_{s,k} = \sum_{j=1}^{k} (-1)^{j-1}\, S_{s,k-j} P_{s,j} \tag{3}$$

With some calculations, one can remove the recursion in (3) and derive the following formula for $S_{s,d}$,

$$S_{s,d} = (-1)^s \sum_{\substack{j_1,\ldots,j_d \geq 0 \\ \sum_{k=1}^{d} k j_k = d}} \prod_{k=1}^{s} \frac{(-1)^{j_k}}{j_k!\, k^{j_k}} P_{s,k}^{j_k}. \tag{4}$$

We want to bound the size of the circuit that computes $S_{s,d}$ according to (4). Polynomials $P_{s,k}^{j_k}$ can be computed by a $\Pi\Sigma\Pi$-circuit of size $O(skj_k) = O(ds^2)$. Hence, for the product of these polynomials, we get size $O(ds^3)$. Crucial for the circuit size is the size of the outer sum. The number of summands in (4) is exactly the *unordered partition number of* d: For unordered partitions, one considers the number of ways to write

$$d = a_1 + a_2 + \cdots + a_d, \quad \text{where } a_1 \geq a_2 \geq \cdots \geq a_d \geq 0. \tag{5}$$

This is the same as the number of ways to write

$$d = j_1 + 2j_2 + \cdots + dj_d, \quad \text{where } j_1, j_2, \ldots, j_d \geq 0.$$

Here j_k denotes the number of terms a_i equal to k in (5).

A known upper bound on the partition number of d is $2^{O(\sqrt{d})}$. Hence, polynomial $S_{s,d}$ can be computed by a $\Sigma\Pi\Sigma\Pi$-circuit of size $2^{O(\sqrt{d})}\,\mathrm{poly}(s)$. □

Remark. We made the assumption that the underlying field $\mathbb{F}$ has characteristic 0. The reason are the denominators occurring in (4). Hence, we could also work with fields of large enough characteristic.

Lemma 4.3 (Set-multilinearization). *Let C be an homogeneous circuit of size s and product-depth Δ that computes an n-variate set-multilinear polynomial over variable sets $(X_1, X_2, \ldots, X_d)$. Then there is a circuit equivalent to C that is* set-multilinear, *of product-depth Δ and size $d^{O(d)}\,\mathrm{poly}(s)$.*

Proof. Let g be a gate in C that computes a polynomial of degree d_g. Because C is homogeneous, all monomials computed at g have the same degree d_g. Our goal is to split gate g into its set-multilinear parts. To do so, for any subset $S \subseteq [d]$ of size $|S| = d_g$, we create a copy g_S of g that computes the set-multilinear part of g restricted to S. This is done inductively in C, starting at the input level.

If g is an input variable $x \in X_i$, for some $i \in [d]$, then $d_g = 1$ and we define $g_{\{i\}} = x$ and $g_{\{j\}} = 0$, for $j \neq i$. Another case is that the input is just a constant c. Then $d_g = 0$ and we define $g_\emptyset = c$.

Now let g be an inner gate, i.e. a $+$-gate or a $\times$-gate.

- Case 1: g is $+$-gate, $g = c_1 g^{(1)} + c_2 g^{(2)} + \cdots + c_r g^{(r)}$, for some $r \geq 1$ and constants $c_1, c_2, \ldots, c_r$. Then, for any $S \subseteq [d]$ with $|S| = d_g$, we define

$$g_S = c_1 g_S^{(1)} + c_2 g_S^{(2)} + \cdots + c_r g_S^{(r)}.$$

- Case 2: g is $\times$-gate, $g = g^{(1)}g^{(2)}\cdots g^{(r)}$, for some $r \geq 1$. Then, for any $S \subseteq [d]$ with $|S| = d_g$, we define

$$g_S = \sum_{\substack{(S_1,S_2,\ldots,S_r) \text{ partition of } S \\ |S_i|=d_{g^{(i)}},\ i=1,2,\ldots,r}} g^{(1)}_{S_1} g^{(2)}_{S_2} \cdots g^{(r)}_{S_r} . \tag{6}$$

Note that the depth of the circuit does not change because the sum in (6) can be merged with the sum-gate in the next level. With respect to circuit size, for a $+$-gate, we get $\binom{d}{d_g} \leq 2^d$ new gates, and for a $\times$-gate, we get

$$\binom{d}{d_g}\binom{d_g}{d_{g_1}, d_{g_2}, \ldots, d_{g_r}} \leq 2^d\, d^d \tag{7}$$

new gates. For the upper bound in (7), we show in Lemma 4.4 below, that the *multinomial coefficient* in (7) is bounded by d^d. This proves the size bound of the lemma. □

Lemma 4.4. *Let* $d = d_1 + d_2 + \cdots d_r$. *Then*

$$\binom{d}{d_1, d_2, \ldots, d_r} \leq d^d .$$

Proof. By the *Multinomial Theorem*, we have

$$\begin{aligned} d^d = (d_1 + d_2 + \cdots + d_r)^d &= \sum_{j_1+j_2+\cdots+j_r=d} \binom{d}{j_1, j_2, \ldots, j_r} d_1^{j_1}\, d_2^{j_2} \cdots d_r^{j_r} \\ &\geq \binom{d}{d_1, d_2, \ldots, d_r} d_1^{d_1}\, d_2^{d_2} \cdots d_r^{d_r} \end{aligned}$$

□

Remark. 1. Recall that set-multilinear circuits are defined as a sub-class of homogeneous circuits. That is, the set-multilinear circuit constructed from a homogeneous circuit in Lemma 4.3 is homogeneous as well.

2. The increase in size in the set-multilinearization step by factor $d^{O(d)}$ is much larger than the factor $2^{O(\sqrt{d})}$ in the homogenization step. If one could improve the factor for set-multilinearization to $2^{O(\sqrt{d})}$ as well, one would get better lower bounds, actually exponential lower bounds according to the LST-Sigact News Guest Column (2022).

Given circuit C with product-depth Δ and size s that computes f, we first apply Lemma 4.1 to C and obtain homogeneous circuit C'. Then we apply Lemma 4.3 to C' and end up with the set-multilinear circuit C'' with product-depth 2Δ and size $d^{O(d)}\,\mathrm{poly}(s)$. The lower bound will be shown for C''. This also yields a lower bound for C.

The key idea of the proof is already present in the case $\Delta = 1$. That is, we prove Theorem 3.2 and show a lower bounds for $\Sigma\Pi\Sigma$-circuits. By the above transformation, this translates to proving good enough lower bound for set-multilinear $\Sigma\Pi\Sigma\Pi\Sigma$-circuits, i.e. of depth 5.

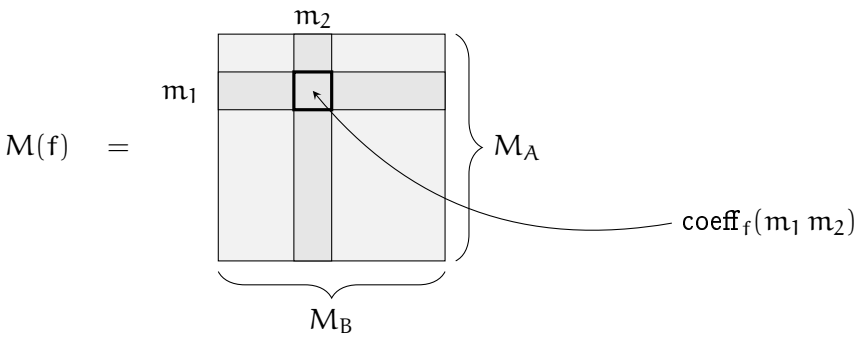
5 The complexity measure

Let f be a set-multilinear polynomial over the sets of variables $(X_1, X_2, \ldots, X_d)$. We define a complexity measure $\mu(f)$ for f.

We split the variables into two parts. Let $A \subseteq [d]$ and $B = [d] - A$. Define the set-multilinear monomials over A and B,

$$M_A = \{\, m \mid m = \prod_{i \in A} x_i, \text{ where } x_i \in X_i \,\},$$
$$M_B = \{\, m \mid m = \prod_{i \in B} x_i, \text{ where } x_i \in X_i \,\}.$$

We define the *coefficient matrix* $M(f)$ for f with respect to the partition A, B. Each monomial from M_A or M_B is assigned to a row or column of $M(f)$, respectively. For $m_1 \in M_A$ and $m_2 \in M_B$, the entry of $M(f)$ at position (m_1, m_2) is the coefficient of the monomial $m_1 m_2$ in f.



The dimension of $M(f)$ is $|M_A| \times |M_B|$. Note that

$$|M_A| = \prod_{i \in A} |X_i| \quad \text{and} \quad |M_B| = \prod_{i \in B} |X_i|.$$

A special case is when one of A, B is empty. We define $M_\emptyset = \{1\}$. Then $|M_\emptyset| = 1$ and $M(f)$ is a column vector when $B = \emptyset$, and a row vector, when $A = \emptyset$.

Remark. The coefficient matrix $M(f)$ is used in various contexts. It was for example used by Nisan (1991) to prove lower bounds for algebraic branching programs. It is also called *partial derivative matrix* or *communication matrix* of f. In automata theory it is generalized to the *Hankel matrix* - not to be confused with the Hankel matrix in Linear Algebra.

We use the rank of $M(f)$ as a complexity measure for f. Actually, it will turn out that it is more convenient to consider a *normalized* version of the rank. In case of a square matrix, one simply divides by the number of rows to normalize the maximum value to 1. However, we also have the case of rectangular matrices. Then we have $\text{rank}(M_f) \leq \min\{|M_A|, |M_B|\}$. Hence, one way to normalize the rank could be to divide by $\min\{|M_A|, |M_B|\}$. However, then we would not have some of the properties we want. In particular, we want the measure to be additive and multiplicative as stated in Lemma 5.1 below. That would not work with a division by $\min\{|M_A|, |M_B|\}$.

Instead, both values, $|M_A|$ and $|M_B|$ should go into the normalization. An idea is to use the *geometric mean* of the dimensions, $\sqrt{|M_A|\,|M_B|}$. Recall that the geometric mean is closer to the minimum than to the maximum:

$$\min\{|M_A|, |M_B|\} \;\le\; \sqrt{|M_A|\,|M_B|} \;\le\; \frac{|M_A| + |M_B|}{2} \;\le\; \max\{|M_A|, |M_B|\}.$$

The complexity measure $\mu(f)$ is defined as the *relative-rank* of $M(f)$, denoted by rel-rank,

$$\mu(f) = \text{rel-rank}(M(f)) = \frac{\text{rank}(M(f))}{\sqrt{|M_A|\,|M_B|}}. \tag{8}$$

Remark. Nisan and Wigderson (1997) worked with a normalized measure as well. It was based on the dimension of the *partial derivative space* of f instead of the rank of the coefficient matrix.

From Linear Algebra rules for the rank, we get that the relative rank additive and multiplicative.

Lemma 5.1 (Properties of μ). *Let f, g be set-multilinear polynomials.*

1. Upper bound. *For any partition A, B of the variable sets,*

$$\mu(f) \le \min\left\{\sqrt{\frac{|M_A|}{|M_B|}}, \sqrt{\frac{|M_B|}{|M_A|}}\right\}.$$

   In particular, $\mu(f) \le 1$.

2. Sub-additivity. *If f and g are defined via the same partition A, B, then $f + g$ is set-multilinear and*

$$\mu(f + g) \le \mu(f) + \mu(g).$$

3. Multiplicativity. *If f and g are defined via disjoint sub-partitions, then fg is set-multilinear and*

$$\mu(fg) = \mu(f)\,\mu(g).$$

For the multiplicativity, note that coefficient matrix $M(fg)$ is a Kronecker product of $M(f)$ and $M(g)$. A common notation is $M(fg) = M(f) \otimes M(g)$. Thereby the rank multiplies.

We mention two more properties of the relative rank that we will use.

1. One reason why we normalized the rank can be observed at a product $q = f_1 f_2 \cdots f_k$. Since all measures are bounded by 1, any $\mu(f_i)$ is an upper bound on $\mu(q)$,

$$\mu(q) \le \mu(f_i), \text{ for all } i.$$

   Hence, this gives an easy way to bound the measure for q.

2. Maybe the most important property for us is that, for rectangular matrices, the measure is strictly less than 1. Consider an $a \times b$ coefficient matrix M, where b is smaller than a, say $b = a^{1-\delta}$, for some $\delta > 0$. Suppose M has full rank b. Then, if we would have defined the

measure by dividing by the shorter side, i.e. b, we would get measure 1. With our above definition (8) of the measure, we get by Lemma 5.1 (1) the bound

$$\sqrt{\frac{a^{1-\delta}}{a}} = \frac{1}{a^{\delta/2}}.$$

This loss in the measure, compared with measure 1, will be crucial in the lower bound argument.

5.1 Example 1: Iterated Matrix Multiplication (IMM)

Given $m \times m$ matrices $X_1, X_2, \ldots, X_d$, for some $m, d \geq 1$, we already defined

$$\mathrm{IMM}_{m,d}(X_1, X_2, \ldots, X_d) = (X_1 X_2 \cdots X_d)_{1,1}.$$

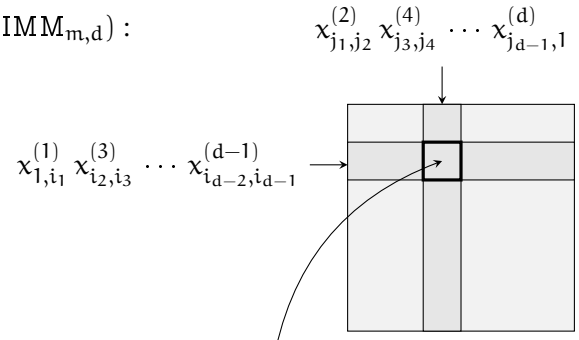
For an explicit formula, let $X_k = \left(x^{(k)}_{i,j}\right)_{1 \leq i,j \leq m}$. Then we have

$$\mathrm{IMM}_{m,d} = \sum_{i_1, i_2, \ldots, i_{d-1} \in [m]} x^{(1)}_{1,i_1} x^{(2)}_{i_1,i_2} x^{(3)}_{i_2,i_3} \cdots x^{(d)}_{i_{d-1},1}$$

Observe that $\mathrm{IMM}_{m,d}$ is set-multilinear of degree d over $n = dm^2$ variables.

We consider the coefficient matrix $M(\mathrm{IMM}_{m,d})$. We assume that d is even and split the variables into odd and even indices. That is, let $A = \{1, 3, \ldots d-1\}$ and $B = \{2, 4, \ldots, d\}$. Now observe that for any monomial $x^{(1)}_{1,i_1} x^{(3)}_{i_2,i_3} \cdots x^{(d-1)}_{i_{d-2},i_{d-1}} \in M_A$, there is precisely one monomial in M_B, namely $x^{(2)}_{i_1,i_2} x^{(4)}_{i_3,i_4} \cdots x^{(d)}_{i_{d-1},1}$, such that the product appears in $\mathrm{IMM}_{m,d}$, and vice versa. Hence, $M(\mathrm{IMM}_{m,d})$ is the identity matrix and therefore has full rank m^d. With respect to the relative rank measure, we conclude that $\mu(\mathrm{IMM}_{m,d}) = 1$.

$M(\mathrm{IMM}_{m,d})$:



$$\mathrm{coeff}\left(x^{(1)}_{1,i_1} x^{(3)}_{i_2,i_3} \cdots x^{(d-1)}_{i_{d-2},i_{d-1}} x^{(2)}_{j_1,j_2} x^{(4)}_{j_3,j_4} \cdots x^{(d)}_{j_{d-1},1}\right) = \begin{cases} 1, & \text{if } (i_1, i_2, \ldots, i_{d-1}) = (j_1, j_2, \ldots, j_{d-1}), \\ 0, & \text{otherwise.} \end{cases}$$

On the other, we show next that any set-multilinear circuit of depth 3 over variables $X_1, X_2, \ldots, X_d$ has small measure μ. That is, we now consider each matrix X_i from above simply as a set of m^2 variables.

Let C be set-multilinear $\Sigma^s\Pi\Sigma$-circuit that computes polynomial f. Let A, B be a partition of the variables.

The sum-gates above the input level of C compute linear polynomials. Since C is set-multilinear, each such linear polynomial ℓ must only involve a single part X_i of the variables. Hence, depending on whether i is in A or B, in the sub-partition for $M(\ell)$, say A_ℓ, B_ℓ, we have $A_\ell = \{i\}$ and $B_\ell = \emptyset$, or vice versa. Recall from the definition that then matrix $M(\ell)$ is either a column or a row vector. Therefore, for $\ell \neq 0$, we have $\operatorname{rank}(M(\ell)) = 1$, and hence

$$\mu(\ell) = \frac{1}{\sqrt{|X_i|}}.$$

Let q be the polynomial computed at a product-gate of C. Because C is and set-multilinear of degree d, exactly d sum gates feed into the product-gate, one for each set of variables X_i, for $i = 1, 2, \ldots, d$. Therefore, by the multiplicativity of μ, we get

$$\mu(q) = \frac{1}{\sqrt{|X_1| \cdots |X_d|}}.$$

Finally, we can use the sub-additivity of μ for the sum at the output-gate of C. It has fan-in s. Hence, we get

$$\mu(f) \leq \frac{s}{\sqrt{|X_1| \cdots |X_d|}}. \tag{9}$$

Recall that $|X_i| = m^2$. Hence, from (9) we get $\mu(f) \leq \frac{s}{m^d}$

Now consider $f = \mathrm{IMM}_{m,d}$. Because $\mu(\mathrm{IMM}_{m,d}) = 1$, we get $s \geq m^d$.

Recall that the number of variables is $n = dm^2$. When we further assume that $d = O(\log n)$, we get

$$m^d = \left(\frac{n}{d}\right)^{\frac{d}{2}} = n^{\Omega(d)}.$$

Theorem 5.2. *Let $d = O(\log n)$. Any set-multilinear $\Sigma^s\Pi\Sigma$-circuit that computes polynomial $\mathrm{IMM}_{m,d}$ has size at least*

$$s \geq m^d = n^{\Omega(d)}.$$

5.2 Example 2: Product of Inner Products (PIP)

The *Product of Inner Products* problem is defined as follows. Given vectors $X_1, X_2, \ldots, X_d$ of length m, for some $m, d \geq 1$. The task is to compute the product of the inner product of subsequent pairs of vectors,

$$\mathrm{PIP}_{m,d}(X_1, X_2, \ldots, X_d) = \prod_{k=1}^{d/2} X_{2k-1} \, X_{2k}.$$

We assume again that d is even. For an explicit formula, let $X_k = \left(x_i^{(k)}\right)_{1 \leq i \leq m}$. Then we have

$$\mathrm{PIP}_{m,d} = \prod_{k=1}^{d/2} \sum_{i=1}^{m} x_i^{(2k-1)} \, x_i^{(2k)} \tag{10}$$

Observe that $\mathrm{PIP}_{m,d}$ is set-multilinear of degree d over $n = dm$ variables.

We consider the coefficient matrix $M(\mathrm{PIP}_{m,d})$. We split the variables again into odd and even indices, $A = \{1, 3, \dots d-1\}$ and $B = \{2, 4, \dots, d\}$. Similar as for IMM, for any monomial $x_{i_1}^{(1)} x_{i_2}^{(3)} \cdots x_{i_{d-1}}^{(d-1)} \in M_A$, there is precisely one monomial in M_B, namely $x_{i_1}^{(2)} x_{i_2}^{(4)} \cdots x_{i_{d-1}}^{(d)}$, such that the product appears in $\mathrm{PIP}_{m,d}$, and vice versa. Hence, also $M(\mathrm{PIP}_{m,d})$ is the identity matrix and $\mu(\mathrm{PIP}_{m,d}) = 1$.

Hence, we get again a lower bound against set-multilinear $\Sigma\Pi\Sigma$-circuits similar as for IMM. However, note that (10) gives $\Pi\Sigma\Pi$-circuit for $\mathrm{PIP}_{m,d}$ of size $O(md)$. So we see that circuits with product-depth 2, i.e. of depth 4, or even $\Pi\Sigma\Pi$-circuits, can have measure 1. It seems as our lower bound technique only works for product-depth 1, i.e. up to set-multilinear $\Sigma\Pi\Sigma$-circuits of depth 3. Recall that we would need lower bounds against set-multilinear circuits of depth 5 in order to get lower bounds for general circuits of depth 3.

Remark. A different technique to prove lower bounds is the *Shifted Partial Derivative Measure* introduced by Kayal (2012). It was used by Gupta, Kamath, Kayal, and Saptharishi (2014) to show a $m^{\Omega(\sqrt{d})}$ lower bound for set-multilinear circuits of depth 4 that compute $\mathrm{IMM}_{m,d}$. Also here it seems as the techniques does not extend to depth 5.

5.3 Summarizing the technique so far and an idea what to change

In the above examples, we start out with a polynomial that is supposed to be hard for the class of circuits under consideration, like IMM or PIP. Note that with these polynomials, we already define the variable sets $X_1, X_2, \dots, X_d$. Also, to get measure one, the polynomials define the partition A, B of the variable sets. All this is fixed when we now try to bound the measure μ for set-multilinear circuits that compute the hard polynomial. Recall that the coefficient matrix, and hence the measure, depends on this setting.

Consider again the $\Pi\Sigma\Pi$-circuit for PIP according to (10). The coefficient matrix of the inner product parts, $M(X_{2k-1}X_{2k})$, is the identity matrix, i.e. a square matrix of full rank. The degree of the monomials is small, only 2. Then the outer product simply multiplies the measure 1 inner product parts to a measure 1 circuit for PIP.

So the problem with the $\Pi\Sigma\Pi$-circuit is that already the inner $\Sigma\Pi$-circuits have measure 1. Note that this is *not* in contradiction with the small measure we proved for depth 3 circuits above: These circuits computed the final polynomial of degree d, whereas here, we consider intermediate gates that compute small degree polynomials, degree 2 in case of PIP. Hence, the question now is: *Is there a way to change the setting, so that the inner $\Sigma\Pi$-circuits have smaller measure?*

Note that the set-up for PIP is nicely tailored: all variable sets X_k have the same size which results in the square matrix $M(X_{2k-1}X_{2k})$ of full rank. A simple way to disturb the full rank would be to define sets X_k of *different sizes* so that the resulting coefficient matrix of the inner $\Sigma\Pi$-circuits is no longer square, but rectangular. Then the rank is bounded by the length of the shorter side of the rectangle, i.e., we get some loss in their rank. This loss in the rank should be significant enough so that the overall circuit has a small measure.

The idea we try to pursue now is as follows: Come up with a setting of variable sets X_k of different sizes, such that with respect to a $\Sigma\Pi\Sigma\Pi\Sigma$-circuit of depth 5, the inner $\Sigma\Pi\Sigma$-circuits have rectangular coefficient matrices, such that the loss in their rank implies a small measure of the

overall circuit of depth 5. Clearly, one also has to adapt the hard polynomial to the new setting. In the next section, we will define such a set-up and show that the above idea indeed works.

Note that the same idea of *unbalancing* coefficient matrices was already used before to obtain lower bounds, for example for multilinear formulas by Raz (2004), or for multilinear circuits by Raz, Shpilka, Yehudayoff (2008).

6 Proof of Theorem 3.2

We define a variable setting and a hard polynomial of measure 1. Then we show that circuits in this setting have small measure, first for depth 3, then for depth 5.

6.1 The hard polynomial

Let $X_1, X_2, \ldots, X_d$ be disjoint sets of variables. For some $t \leq d$, we split them at t into

$$A = [t] \quad \text{and} \quad B = [d] - [t].$$

The variable sets have different sizes. The size of the A-sets we denote by m which is a power of 2, i.e. $m = 2^k$, for some k. The size of the B-sets is $m^{1-\delta}$, for $\delta = \frac{1}{2\sqrt{d}}$, and should be a power of 2 as well, $m^{1-\delta} = 2^\ell$, for some ℓ.

$$|X_i| = \begin{cases} m = 2^k, & \text{for } i \in A, \\ m^{1-\delta} = 2^{k(1-\frac{1}{2\sqrt{d}})} = 2^\ell, & \text{for } i \in B. \end{cases} \tag{11}$$

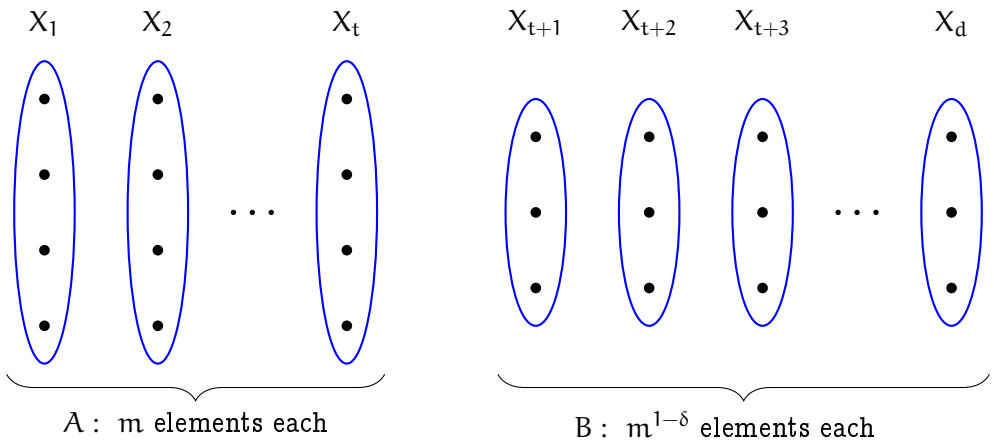


Figure 1: *The variables are split into two groups* A *and* B*, where* A *has a smaller number* $t < d/2$ *of sets of larger size* m *and* B *has a larger number* $d - t > d/2$ *of sets of smaller size* $m^{1-\delta}$*. The parameters are chosen such that the number of set-multilinear monomials build from* A *and* B *are the same, i.e.* $|M_A| = |M_B|$.

By n we denote again the total number of variables. Note that $n \leq md$.

Actually, because of the square root involved in δ, we may need to round the expression to define ℓ. For simplicity, we ignore such issues for now. Because we want $\ell = k(1 - \frac{1}{2\sqrt{d}}) < k$, we should have $k \geq 2\sqrt{d}$. Since $k = \log m$, this implies $d \leq \frac{\log^2 m}{4}$. Hence, the construction implies a bound on the degree in terms of the number of variables.

Because the set sizes are powers of 2, we can enumerate the variables by 0-1-words of length k and ℓ, respectively. That is, let

$$X_i = \{ x_\alpha^{(i)} \mid \alpha \in \{0,1\}^k \}, \quad \text{for } i \in A,$$
$$X_i = \{ x_\beta^{(i)} \mid \beta \in \{0,1\}^\ell \}, \quad \text{for } i \in B.$$

We choose the splitting point t such that the coefficient matrix is a square matrix. We have $|M^A| = m^t = 2^{kt}$ and $|M^B| = m^{(1-\delta)(d-t)} = 2^{\ell(d-t)}$. Hence, we choose t such that

$$kt = \ell(d - t).$$

So we define $t = \frac{d\ell}{k+\ell}$. Again, we would have to round the expression to be precise.

The point of the somewhat involved setup so far is that we will show below that any polynomial computed by a set-multilinear circuit of depth 5 has small measure μ. To get a lower bound, we need an explicit polynomial with high measure. The polynomial we define is a *modified IMM* kind of polynomial, denoted mIMM.

$$\mathrm{mIMM}_{m,d} = \sum_{\alpha_1 \cdots \alpha_t = \beta_1 \cdots \beta_{d-t} \in \{0,1\}^{kt}} x_{\alpha_1}^{(1)} \cdots x_{\alpha_t}^{(t)} x_{\beta_1}^{(t+1)} \cdots x_{\beta_{d-t}}^{(d)}, \tag{12}$$

where the sum is over all $\alpha_1, \ldots, \alpha_t \in \{0,1\}^k$ and $\beta_1, \ldots, \beta_{d-t} \in \{0,1\}^\ell$. Note that once $\alpha_1, \ldots, \alpha_t$ are chosen, then $\beta_1, \ldots, \beta_{d-t} \in \{0,1\}^\ell$ are uniquely determined to get a monomial from $\mathrm{mIMM}_{m,d}$, and vice-versa. Therefore $M(\mathrm{mIMM}_{m,d})$ is a permutation matrix which has full rank. Hence, we have $\mu(\mathrm{mIMM}_{m,d}) = 1$.

Remark.

1. Formally, one can show that $\mathrm{mIMM}_{m,d}$ is a projection of $\mathrm{IMM}_{m,d}$. This can be seen by representing the polynomials as *arithmetic branching programs (ABP)*. For IMM one can construct the obvious ABP. For mIMM it is slightly more tricky. One has to interleave the α-variable with β-variables appropriately to get a poly-size ABP. It follows that a lower bound for $\mathrm{mIMM}_{m,d}$ implies the same lower bound for $\mathrm{IMM}_{m,d}$.

2. The exposition here is simplified because we skip the rounding of some of the expressions as mentioned above. If one does precise calculations, i.e. by considering rounded numbers, matrix $M(\mathrm{mIMM}_{m,d})$ might no longer be a square matrix and one has to adapt the definition of $\mathrm{mIMM}_{m,d}$ in (12) because the α-string might have a different length than the β-string. Still one can show that $\mu(\mathrm{mIMM}_{m,d})$ is close enough to 1 to obtain the desired lower bound. The original paper [LST21] has all the details.

6.2 Lower bounds for set-multilinear ΣΠΣ-circuits

Already in Section 5.1, we showed how to bound the measure for ΣΠΣ-circuits. This works very similar in the new variable setting.

Let $X_1, X_2, \ldots, X_d$ be sets of variables and A, B be a partition as defined in the previous section, and let C be a set-multilinear $\Sigma^s\Pi\Sigma$-circuit over these variables that computes some polynomial f.

By the same arguments as in Section 5.1, we get again Equation (9),

$$\mu(f) \leq \frac{s}{\sqrt{|X_1| \cdots |X_d|}}.$$

Recall Equation (11) that we defined the sets X_i such that $|X_i| \geq m^{1-\delta}$, for $i = 1, 2, \ldots, d$, where $\delta = \frac{1}{2\sqrt{d}}$. Plugging this into (9), we get

$$\mu(f) \leq \frac{s}{\sqrt{|X_1| \cdots |X_d|}} \leq \frac{s}{m^{\frac{d}{2}(1-\frac{1}{2\sqrt{d}})}} \leq \frac{s}{m^{\frac{d}{4}}}. \tag{13}$$

We apply the bound (13) to polynomial $f = \mathrm{mIMM}_{m,d}$ defined in Section 6.1. Recall that $\mu(\mathrm{mIMM}_{m,d}) = 1$.

Theorem 6.1. *Any set-multilinear $\Sigma^s\Pi\Sigma$-circuit that computes polynomial $\mathrm{mIMM}_{m,d}$ has size at least*

$$s \geq m^{\frac{d}{4}}. \tag{14}$$

Recall that the number of variables is $n \leq md$ and we assume $d = O(\log n)$. When we write the bound given in (14) in n instead of m, we get

$$s \geq \left(\frac{n}{d}\right)^{\frac{d}{4}} = n^{\Omega(d)}.$$

6.3 Lower bounds for set-multilinear $\Sigma\Pi\Sigma\Pi\Sigma$-circuits

Let $X_1, X_2, \ldots, X_d$ be sets of variables and A, B be a partition as defined in Section 6.1, and let C be a set-multilinear $\Sigma\Pi\Sigma\Pi\Sigma$-circuit over these variables that computes some polynomial f.

Let q be one of the polynomials that is computed at a product gate below the output $\sum$-gate of C. That is, polynomial q is computed by a $\Pi\Sigma\Pi\Sigma$-circuit. Let

$$q = f_1 \cdots f_k,$$

where each f_i is computed by a $\Sigma^{s_i}\Pi\Sigma$-circuit. Recall that all the component circuits are also set-multilinear. Still, the f_i's can have different degrees. But we know that $\sum_{i=1}^{k} \deg(f_i) = \deg(q) = d$.

The high-level idea. In case of a $\Sigma\Pi\Sigma$-circuit as in Section 6.2, the f_i's would already be the bottom layer of linear polynomials and the coefficient matrices $M(f_i)$ would have small rank, namely rank 1. In our case now, the f_i's are themselves computed by $\Sigma\Pi\Sigma$-circuits, and it seems hard to say anything useful about $M(f_i)$ and its rank.

- An easy case is when one of the f_i's has small measure. By (13), this is when some f_i has a high degree. Now we see the advantage of working with the relative rank instead of the rank: Since the μ-values are ≤ 1, any single $\mu(f_i)$ is an upper bound on $\mu(q)$. Now this single sub-circuit will give already a good enough upper bound on $\mu(q)$

- The hard case is when all f_i's have small degree. Recall that we set parameters such that the coefficient matrix $M(f)$ is a square matrix. The matrices $M(f_i)$ have smaller dimensions on sub-partitions of $(X_1, X_2, \ldots, X_d)$. Recall that some of the X_i's have size $m^{1-\delta}$.

  The main point now is that matrices $M(f_i)$ turn out to be *rectangular*. It is not possible for them to be square when the f_i's have small degree. The simple fact that the rank of $M(f_i)$ is bounded by the shorter side of the rectangle gives a good enough loss of the rank for us with respect to the longer side. This will imply a small relative rank.

The details. In order to get a bound on the relative rank of q, we distinguish two cases according to the degrees of the f_i's. Without loss of generality, let f_1 have the largest degree among $f_1, \ldots, f_k$.

Case 1: $\deg(f_1) = d_1 \geq \sqrt{d}$. In this case, it suffices to consider the sub-circuit for f_1 for a size bound. We do a similar calculation as in Section 6.2.

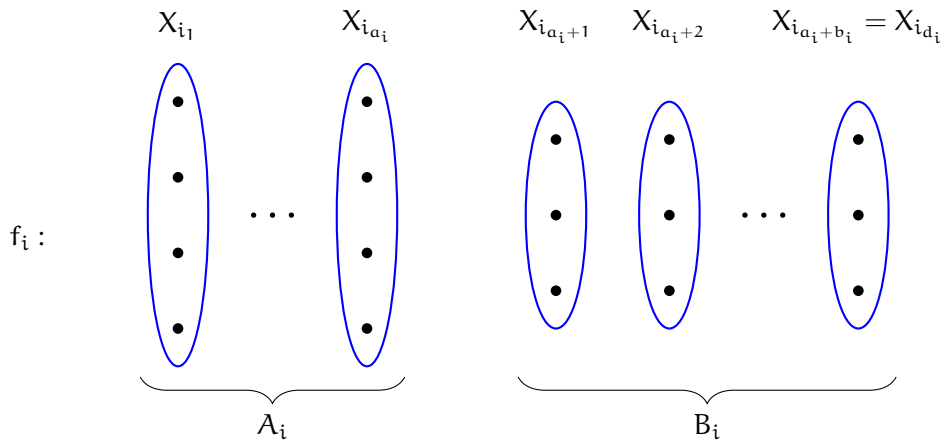
Let f_1 be set-multilinear over the sub-partition $(X_{i_1}, \ldots, X_{i_{d_1}})$. Similar as in Equation (9) and (13) we get

$$\mu(f_1) \leq \frac{s_1}{\sqrt{|X_{i_1}| \cdots |X_{i_{d_1}}|}} < \frac{s_1}{m^{\frac{\sqrt{d}}{2}(1-\frac{1}{2\sqrt{d}})}} \leq \frac{s_1}{m^{\frac{\sqrt{d}}{4}}}.$$

Because μ is multiplicative, we have $\mu(q) = \prod_{i=1}^{k} \mu(f_i)$. Since the μ-values are ≤ 1, we get

$$\mu(q) \leq \mu(f_1) \leq \frac{s_1}{m^{\frac{\sqrt{d}}{4}}}. \tag{15}$$

Case 2: $\deg(f_1) < \sqrt{d}$. Each f_i is set-multilinear over a sub-partition $(X_{i_1}, X_{i_2}, \ldots, X_{i_{d_i}})$, for all $i \in [k]$. In the sub-partition of f_i, let $A_i \subseteq A$ be the sets from A, i.e. of size m, and $B_i \subseteq B$ be the number of sets from B, i.e. of size $m^{1-\delta}$. Let $a_i = |A_i|$ and $b_i = |B_i|$. Then $a_i + b_i = \deg(f_i) < \sqrt{d}$.



The coefficient matrix $M(f_i)$ has $|M_A| = m^{a_i}$ rows and $|M_B| = (m^{1-\delta})^{b_i} = m^{b_i(1-\delta)}$ columns, for all $i \in [k]$. We apply the upper bound $\mu(f_i) \leq \min\left\{\sqrt{\frac{|M_{A_i}|}{|M_{B_i}|}}, \sqrt{\frac{|M_{B_i}|}{|M_{A_i}|}}\right\}$ given in Lemma 5.1 (1).

1. If $|M_{A_i}| \leq |M_{B_i}|$, then
$$\mu(f_i) \leq \sqrt{\frac{|M_{A_i}|}{|M_{B_i}|}} = \frac{1}{m^{(b_i(1-\delta)-a_i)/2}}.$$
2. If $|M_{A_i}| > |M_{B_i}|$, then
$$\mu(f_i) \leq \sqrt{\frac{|M_{B_i}|}{|M_{A_i}|}} = \frac{1}{m^{(a_i-b_i(1-\delta))/2}}.$$

In summary of the two cases, we get

$$\mu(f_i) \leq \frac{1}{m^{|a_i-b_i(1-\delta)|/2}}. \tag{16}$$

The maximum value of the right hand side in (16) is when the exponent $|a_i-b_i(1-\delta)| = |a_i-b_i+b_i\delta|$ is minimal. Recall that $b_i < \sqrt{d}$ and $\delta = \frac{1}{2\sqrt{d}}$. Hence, we have $0 \leq b_i\delta < 1/2$. It follows that the exponent $|a_i - b_i + b_i\delta|$ is minimal for $a_i = b_i = \deg(f_i)/2$. In this case, we have

$$\frac{1}{2}|a_i - b_i + b_i\delta| = \frac{1}{2}b_i\delta = \frac{1}{2}\frac{\deg(f_i)}{2}\frac{1}{2\sqrt{d}} = \frac{\deg(f_i)}{8\sqrt{d}}. \tag{17}$$

Hence, by (16), we get

$$\mu(f_i) \leq \frac{1}{m^{\frac{\deg(f_i)}{8\sqrt{d}}}}.$$

Equation (17) is the core of the whole argument. If the exponent $\frac{1}{2}|a_i - b_i + b_i\delta|$ could be 0, we would only get a trivial and useless bound of $\frac{1}{m^0} = 1$ on $\mu(f_i)$. By (17), the exponent is in a distance to 0 that depends on d and the degree of f_i. This is due to the different sizes of the variable sets X_j and the choice of the other parameters. As a consequence, the relative rank of the f_i's smaller than 1.

By multiplicativity, we get

$$\mu(q) = \prod_{i=1}^{k}\mu(f_i) \leq \frac{1}{m^{\frac{d}{8\sqrt{d}}}} = \frac{1}{m^{\frac{\sqrt{d}}{8}}}. \tag{18}$$

The bounds (15) and (18) we get from the two cases, together with the additivity property of μ yield

$$\mu(f) \leq \frac{s}{m^{\frac{\sqrt{d}}{8}}}, \tag{19}$$

where s is the size of C.

We apply the bound (19) to $f = \mathrm{mIMM}_{m,d}$. Recall that $\mu(\mathrm{mIMM}_{m,d}) = 1$.

Theorem 6.2. *Let d, m and $n \leq dm^2$ such that $d = O(\log n)$. Any set-multilinear $\Sigma\Pi\Sigma\Pi\Sigma$-circuit that computes polynomial $\mathrm{mIMM}_{m,d}$ has size s at least*

$$s \geq m^{\frac{\sqrt{d}}{8}} = n^{\Omega(\sqrt{d})}.$$

To get a lower bound for general ΣΠΣ-circuits, we have to reverse the transformations from Lemmas 4.1 and 4.3. The lemmas increase the size of the circuit by a factor $d^{O(d)}$. Note that for $d = O(\log n)$, from $sd^{O(d)} \geq n^{\Omega(\sqrt{d})}$ it still follows that $s \geq n^{\Omega(\sqrt{d})}$.

Corollary 6.3. *Let* d, m *and* $n \leq dm^2$ *such that* $d = O(\log n)$. *Any* ΣΠΣ*-circuit that computes polynomial* $\mathrm{mIMM}_{m,d}$ *has size* s *at least*

$$s \geq n^{\Omega(\sqrt{d})}.$$

The lower bounds for larger depth circuits proceeds along similar lines, with some technical changes. Observe that in the above argument, at least in the second case, we did not really use that the f_i's are computed by ΣΠΣ-circuits, i.e. in depth 3. A similar argument works when f_i's are computed by larger depth circuits.

7 Follow-up work

The same authors, Limaye, Srinivasan and Tavenas (2022) showed limits of the current technique: It is not possible to show lower bounds of the form $m^{d^{\frac{1}{\mathrm{poly}(\Delta)}}}$ by this technique.

Deepanshu Kush and Shubhangi Saraf (2022) showed slightly improved lower bounds for another class of polynomials, often called the *Nisan-Wigderson polynomials*. They follow in part the technique presented here. However, they return to equal size variable sets X_k and instead choose the partition sets A and B uniformly at random.

Motivated by the LST-technique, Amireddy, Garg, Kayal,Saha, Thankey (2022) showed that the same bounds can also be derived via the shifted partial derivative method. They observe that the unbalancing technique can be carried over to the shifted partial derivative setting in some sense. An interesting aspect of their work is that they work directly with the homogenized polynomial and skip the set-multilinearization step. This results in slightly improved lower bounds.

References

[LST21] Nutan Limaye, Srikanth Srinivasan, and Sébastien Tavenas. Superpolynomial lower bounds against low-depth algebraic circuits. In *62nd Symposium on Foundations of Computer Science, FOCS*, pages 804–814. IEEE, 2021.