Set-multilinear and non-commutative formula lower bounds for iterated matrix multiplication

Nutan Limaye, Srikanth Srinivasan, Sébastien Tavenas

Set-Multilinear and Non-commutative Formula Lower Bounds for Iterated Matrix Multiplication

Nutan Limaye [*1], Srikanth Srinivasan[2], and Sébastien Tavenas [†3]

[1]ITU Copenhagen
[2]Aarhus University
[3]Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, LAMA

Abstract

An Algebraic Formula for a polynomial $P \in \mathbb{F}[x_1, \ldots, x_N]$ is an algebraic expression for $P(x_1, \ldots, x_N)$ using variables, field constants, additions and multiplications. Such formulas capture an algebraic analog of the Boolean complexity class NC^1. Proving lower bounds against this model is thus an important problem.

It is known that, to prove superpolynomial lower bounds against algebraic formulas, it suffices to prove good enough lower bounds against restricted kinds of formulas known as Set-Multilinear formulas, for computing a polynomial $P(x_1, ..., x_N)$ of degree $O(\frac{\log N}{\log \log N})$. In the past, many superpolynomial lower bounds were found, but they are of the form $\Omega(f(d)\,\mathrm{poly}(N))$ (where f is typically a subexponential function) which is insufficient to get lower bounds for general formulas. Recently, the authors proved [13] the first *non-FPT* lower bounds, i.e., a lower bound of the form $N^{\Omega(f(d))}$, against small-depth set-multilinear formulas (and also for circuits). In this work, we extend this result in two directions.

1. **Large-depth set-multilinear formulas.** In the setting of general set-multilinear formulas, we prove a lower bound of $(\log n)^{\Omega(\log d)}$ for computing the Iterated Matrix Multiplication polynomial $\mathrm{IMM}_{n,d}$. In particular, this implies the first superpolynomial lower bound against unbounded-depth set-multilinear formulas computing $\mathrm{IMM}_{n,n}$.

   As a corollary, this also resolves the homogeneous version of a question of Nisan (STOC 1991) regarding the relative power of Algebraic formulas and Branching programs in the non-commutative setting.

2. **Stronger bounds for homogeneous non-commutative small-depth circuits.** In the small-depth *homogeneous non-commutative* case, we prove a lower bound of $n^{d^{1/\Delta}/2^{O(\Delta)}}$, which yields non-FPT bounds for depths up to $o(\sqrt{\log d})$. In comparison, the bound in [13] works in the harder *commutative set-multilinear* setting, but only up to depths $o(\log \log d)$. Moreover, our lower bound holds for all values of d, as opposed to the set-multilinear lower bound of [13], which holds as long as d is small, i.e., $d = O(\log n)$.

[*]Funded by SERB project File no. MTR/2017/000909.

[†]Funded by the project BIRCA from the IDEX of Univ. Grenoble Alpes (Contract 183459). The author also benefited from an accommodation paid by IIT Bombay during a visit of the other two authors.

1 Introduction

Polynomials and Algebraic Formulas. Let $P(x_1, \ldots, x_N)$ be a polynomial over a field $\mathbb{F}$. An *Algebraic Formula* computing a polynomial $P(x_1, \ldots, x_N)$ is a directed tree in which the leaves are labelled by variables from $X = \{x_1, \ldots, x_N\}$ or field constants and the internal nodes are labelled by addition or multiplication operators. If an internal node v is labelled by the addition operator, then it computes a linear combination of its inputs u_i, where the coefficients of the linear combinations are field elements labelling the edges between the u_is and v. Similarly, if it is labelled by the multiplication operator, then it computes the product of its inputs. The *size* of the formula is the number of nodes in the tree. The *product-depth* of the formula is the largest number of multiplication gates along any root-to-leaf path.

The class of polynomials that have algebraic formulas of polynomial size is denoted VP_e (or sometimes VF) and is an algebraic analog of the Boolean complexity class NC^1. Proving lower bounds for this class is therefore an important problem in complexity theory in general, and in particular for *Algebraic Complexity theory*, which is the study of the computational complexity of algebraic problems of this kind (see, e.g. [2, 21, 19] for nice introductions to this area). The problem has been investigated for many years and we have several lower bounds against many restricted classes of formulas [1, 8, 15, 17, 5, 6, 11, 13].

Set-multilinear formula lower bounds. Recall that a polynomial $P(x_1, \ldots, x_N)$ is homogeneous if each monomial has the same total degree. It is multilinear if every variable occurs at most once in any monomial. Suppose the underlying variable set is partitioned into d sets, $X_1, \ldots, X_d$, then the polynomial is set-multilinear with respect to this variable partition if each monomial in P has exactly one variable from each set.

Many interesting and well-studied polynomials are in fact set-multilinear. For example, the Determinant and the Permanent polynomials, which are central to algebraic complexity theory, are set-multilinear (w.r.t. *the row variables*). Another well-studied polynomial, namely the Iterated Matrix Multiplication polynomial, is also set-multilinear. As we will use this polynomial in what comes next, we recall the definition. The Iterated Matrix Multiplication polynomial $\mathrm{IMM}_{n,d}$ is defined on $N = dn^2$ variables, where the variables are partitioned into d sets $X_1, \ldots, X_d$ of size n^2, each of which is represented as an $n \times n$ matrix with distinct variable entries. The polynomial $\mathrm{IMM}_{n,d}$ is defined to be the polynomial that is the $(1,1)$th entry of the product matrix $X_1 \cdot X_2 \cdots X_d$.

Corresponding to the above variants of polynomials classes, we also define different models of computation. An algebraic formula is set-multilinear with respect to a variable partition $(X_1, \ldots, X_d)$ if each internal node in the formula computes a set-multilinear polynomial (w.r.t. a variable partition $(X_j : j \in I)$ where I is a subset of $[d]$). Similarly, we define homogeneous formulas and multilinear formulas.

We have several interesting lower bound results against set-multilinear formulas. Nisan and Wigderson [15] proved the first exponential lower bound for product-depth 1 set-multilinear formulas. In particular, among other results, they proved that any product-depth 1 set-multilinear formula computing $\mathrm{IMM}_{n,d}$ must have size $n^{\Omega(d)}$. They introduced the *partial derivative technique* to prove this lower bound. Building on this technique, Raz [17] showed the first super-polynomial lower bound on the size of any arbitrary depth multilinear formula (of arbitrary depth) computing the $n \times n$ Determinant and Permanent. In follow-up works [16, 5] other candidate multilinear polynomials, of varying complexity, were shown to be hard for arbitrary depth polynomial sized multilinear formulas.

Given the large array of lower bounds for restricted models of computation and lack thereof for general formulas, the following question arises naturally. Can we use the known lower bounds

for restricted classes of formulas to obtain lower bounds for general formulas?

An intriguing observation of Raz [18] suggests a way. Raz showed that if a set-multilinear polynomial of degree d has an algebraic formula of size s, then it also has a *set-multilinear* formula of size $\text{poly}(s) \cdot (\log s)^{O(d)}$. In particular, for a set-multilinear polynomial P of degree $d = O(\log N / \log\log N)$, it follows that P has a formula of size $\text{poly}(N)$ if and only if P has a set-multilinear formula of size $\text{poly}(N)$.

This offers us a possible route towards proving general algebraic formula lower bounds via 'hardness escalation' from the set-multilinear case. It is not hard to show that a *random* polynomial of degree d in N variables has no formulas of size $N^{o(d)}$. If we could prove such a lower bound against set-multilinear formulas for computing an explicit polynomial of small degree d, then we would be done.

Non-FPT lower bounds. On the one hand, we have several lower bounds for set-multilinear formulas and on the other hand we have the hardness escalation result by Raz. The missing piece of the puzzle has been the quality of the lower bound needed for making escalation possible. Specifically, all the lower bounds known for arbitrary depth set-multilinear formulas are of the form $\Omega(f(d)\,\text{poly}(N))$ (where f is typically a superpolynomial but at best exponential function). Using an analogy to Parameterized Complexity Theory [4], we call such bounds *FPT bounds.* However, we would like to prove $N^{\omega_d(1)}$ lower bounds. We call such bounds *non-FPT bounds.*

While we do not have non-FPT bounds for general set-multilinear formulas, we have such bounds in the small product-depth setting. The result by Nisan and Wigderson, mentioned above, is the first example of such a bound. Recently, the authors generalised that result [13] and obtained non-FPT bounds for all constant product-depth formulas (and also for circuits)[1]. This result also gave interesting consequences. Specifically, it gave the first superpolynomial lower bounds against *all* constant depth formulas.

Our Results. In this work, we take a step towards proving non-FPT lower bounds for arbitrary depth set-multilinear formulas. Indeed, we prove a lower bound which is already exponential in some $\omega_d(1)$ and where the basis of the exponent increases with n. Specifically, we prove the following theorem.

Theorem 1. *Let n, d, Δ be growing parameters with $\Delta \leqslant O(\log d)$. Then any set-multilinear formula of product-depth at most Δ for $\text{IMM}_{n,d}$ must have size at least $(\log n)^{\Omega(\Delta d^{1/\Delta})}$. Further, any set-multilinear formula for $\text{IMM}_{n,d}$ must have size at least $(\log n)^{\Omega(\log d)}$.*

To put the above result in context, this improves the above mentioned result of Nisan and Wigderson, who proved a lower bound of $\exp(\Omega(d^{1/\Delta}))$, it can be improved to $\exp(\Omega(\Delta d^{1/\Delta}))$ with a slightly more careful analysis. On the other hand, the standard divide-and-conquer strategy for constructing formulas for $\text{IMM}_{n,d}$ yields an upper bound of $n^{O(\Delta d^{1/\Delta})}$. If we conjecture that this is tight, then we expect to change the base of the exponent in the lower bound to be n. Our result seems to be the first to achieve any dependence on n.

The above result also extends the results from [13] in two ways.

- It gives a lower bound for $\text{IMM}_{n,d}$ against formulas of *any* depth. The result in [13] works for depths up to $o(\log\log d)$.
- It puts no restriction on the degree of the polynomial. The lower bound in [13] essentially works for degree $d = O(\log n)$.[2]

[1] This result holds for super-constant but small depths as well.

[2] One can also get lower bounds for larger degrees by reducing from the case of $d = O(\log n)$. However, we do not get an improved lower bound as the degree increases.

In the case that $d = n^{\Omega(1)}$, we get a superpolynomial bound of $n^{\Omega(\log \log n)}$ for set-multilinear formulas computing $\mathrm{IMM}_{n,d}$.

Corollary 2. *Any set-multilinear formula computing* $\mathrm{IMM}_{n,n}$ *must have size at least* $n^{\Omega(\log \log n)}$.

Notice that Raz's escalation result is only useful when we have a non-FPT lower bound for $d = O(\log N / \log \log N)$. But for small d (say $d \leqslant (\log n)^{O(1)}$), our result yields a bound of less than n, which is trivial. This may indicate that the above is not really useful from a hardness escalation perspective.

However, we observe that there is an interesting connection. Here, notice that our lower bound is for $\mathrm{IMM}_{n,n}$, which is a *self-reducible* polynomial. In this context, this refers to the fact that we can construct formulas for $\mathrm{IMM}_{n,n}$ by recursively using formulas for $\mathrm{IMM}_{n,d}$ (for any $d < n$). In particular, if we had formulas of size $n^{o(\log d)}$ for $\mathrm{IMM}_{n,d}$, this would imply formulas of size $n^{o(\log n)}$ for $\mathrm{IMM}_{n,n}$. Stated in the contrapositive, this means that an optimal $n^{\Omega(\log n)}$ lower bound for $\mathrm{IMM}_{n,n}$ implies non-FPT lower bounds for $d < n$, which would then imply general formula lower bounds via escalation. Our superpolynomial lower bound makes the first non-trivial progress towards this goal for unbounded depth set-multilinear formulas (but see also Related work below).

The above corollary also implies an interesting result for *Non-commutative Algebraic Formulas.* A non-commutative algebraic formula is defined just as a standard algebraic formula, except that the underlying variables do not commute.[3] In this setting, explicit superpolynomial formula lower bounds (and also non-FPT lower bounds) follow easily from work of Nisan [14]. However, it is still interesting to prove a superpolynomial lower bound against non-commutative formulas computing $\mathrm{IMM}_{n,d}$, as this would separate the complexity classes VBP[4] and VP_e in the non-commutative setting, which would solve an open question due to Nisan [14]. Our result, along with the fact that non-commutative homogeneous formulas yield set-multilinear formulas (see Lemma 6), implies that we have resolved Nisan's question for homogeneous formulas.

Corollary 3. *Any non-commutative homogeneous formula computing* $\mathrm{IMM}_{n,n}$ *must have size at least* $n^{\Omega(\log \log n)}$.

As mentioned above, the standard divide-and-conquer strategy gives set-multilinear formulas of product-depth Δ and size $n^{O(\Delta d^{1/\Delta})}$ computing $\mathrm{IMM}_{n,d}$. This means that we can potentially prove a lower bound of $n^{\Omega(\Delta d^{1/\Delta})}$ in this setting. Unfortunately, we do not know how to prove this kind of bound even for product-depth three[5].

Here, we prove such a lower bound in the case of non-commutative homogeneous constant-depth formulas. Specifically, we prove the following theorem.

Theorem 4. *Let* n, d *be any growing parameters. Any constant non-commutative homogeneous product-depth* Δ *formulas for* $\mathrm{IMM}_{n,d}$ *must have size* $n^{\Omega(d^{1/\Delta})}$. *Further, if* Δ *is any growing parameter, then any product-depth* Δ *non-commutative homogeneous formula for* $\mathrm{IMM}_{n,d}$ *must have size at least* $n^{d^{1/\Delta}/2^{O(\Delta)}}$.

As any circuit of product-depth Δ *and size* s *can be converted into a formula of product-depth* Δ *and size* $s^{O(\Delta)}$, *the same lower bounds continue to hold for homogeneous non-commutative circuits.*

[3] Equivalently, the multiplications take place in the non-commutative polynomial ring $\mathbb{F}\langle x_1, \ldots, x_N \rangle$.

[4] This is the algebraic analogue of the complexity class NL and is defined to be the class of polynomials that have efficient *Algebraic Branching Programs.* $\mathrm{IMM}_{n,n}$ is the canoncial complete problem for this complexity class, in both the commutative and non-commutative settings.

[5] As mentioned above, the best lower bound we know when d is small in this setting against set-multilinear formulas is $n^{d^{\exp(-O(\Delta))}}$ due to [13]. Even for product-depth 3, this does not get the *right* bound of $n^{\Omega(d^{1/3})}$.

Note that, this gives non-FPT lower bounds for product-depths all the way to $O(\sqrt{\log d})$. Moreover, we can also easily compute such polynomials if we consider formulas of product-depth $\Delta + 1$ (instead of Δ). It thus implies an exponential separation between non-commutative formulas of different product-depth (see Corollary 17).

Related Work. We have had set-multilinear formula lower bounds for unbounded-depth formulas since the early 2000s. Raz [17] showed an $n^{\Omega(\log n)}$ lower bound on the size of any multilinear (and hence in particular set-multilinear) formula for the Determinant, and Dvir, Malod, Perifel and Yehudayoff [5] proved similar lower bounds for another multilinear polynomial on n variables. Unfortunately, these polynomials are not self-reducible in the same sense as IMM is, and so it is unclear if we can use these bounds to obtain non-FPT lower bounds.

However, both these families of polynomials lie in the complexity class VBP. Given that $\mathrm{IMM}_{n,n}$ is complete for this class, one might expect that we get a similar lower bound for this polynomial. Curiously, however, these results do not imply any lower bound for $\mathrm{IMM}_{n,n}$. This is because the underlying reductions to $\mathrm{IMM}_{n,n}$ destroy the multilinearity of the formula, and hence the multilinear formula lower bounds no longer apply. Therefore, as far as we know, despite this progress, we did not have any unbounded-depth formula lower bounds for $\mathrm{IMM}_{n,n}$, even in the set-multilinear setting, prior to this work.

Nisan and Wigderson [15] proved a non-FPT bound for a special kind of commutative set-multilinear formulas called *pure circuits*[6]. More recently, Chatterjee [3] also proved a non-FPT bound for another variety of restricted non-commutative formulas, called *Abecedarian formulas*. Theorem 4 above strengthens the first result.[7] However, it seems incomparable to the second result, as the lower bound result of [3] is for a non-set-multilinear polynomial. At a higher level, our techniques are also quite different, as we use rank-based arguments, while Chatterjee's lower bound uses a result of Hrubeš and Yehudayoff [7] that uses the *sparsity* (number of monomials) in the underlying polynomial. Such a technique does not seem to be applicable in our setting.

Our Techniques. At a high level, the proof outline for our lower bounds in Theorem 1 and Theorem 4 looks very similar to that of many known lower bounds. We design a measure $\mu(\cdot)$ on the space of polynomials such that the measure is *small* for all the polynomials computed by set-multilinear formulas, whereas it is *large* for the IMM polynomial.

Nisan and Wigderson defined such a measure, often called the *partial derivative measure*, and used it to prove the first non-FPT lower bound for product-depth 1 set-multilinear formulas, which we mentioned earlier. The measure has been extended in many ways over the years to obtain many strong lower bounds (see for instance [16, 20, 9, 6, 10]).

We describe the measure here, as it will help in the proof outline. Let $\mathcal{P}, \mathcal{N}$ be a partition of $[d]$. The variable sets $(X_1, \ldots, X_d)$ are partitioned into the *positive variable sets* $(X_i : i \in \mathcal{P})$ and *negative variable sets* $(X_i : i \in \mathcal{N})$. Let us fix this variable partition. Let $\mathcal{M}^{\mathcal{P}}$ be the set of set-multilinear monomials over the positive variable sets and let $\mathcal{M}^{\mathcal{N}}$ be the set of set-multilinear monomials over negative variable sets. Given a set-multilinear polynomial f over $(X_1, \ldots, X_d)$, we define a matrix M_f to be a matrix whose rows are labelled by $\mathcal{M}^{\mathcal{P}}$ and columns are labelled by $\mathcal{M}^{\mathcal{N}}$. For $m_1 \in \mathcal{M}^{\mathcal{P}}$ and $m_2 \in \mathcal{M}^{\mathcal{N}}$, the (m_1, m_2)th entry is the coefficient of $m_1 \cdot m_2$ in f. The measure is defined as the rank of M_f.[8].

[6] Lagarde, Limaye and Srinivasan [12] proved a non-FPT bound for a special kind of non-commutative formulas that are called *Unique Parse Tree* (UPT) formulas, which essentially reproves Nisan and Wigderson's lower bound for pure circuits.

[7] Our quantitative bound is weaker, but the model is stronger.

[8] For the sake of simplicity of exposition, we consider the matrix rank measure here. In the proof we use a slightly different measure called the *relative rank*. We define that formally in Section 2.

The non-commutative homogeneous case. Indeed, Theorem 4 strengthens the lower bound proved in [13] in the non-commutative homogeneous setting. At a conceptual level, the proof of Theorem 4 is similar to the proof of the lower bound in [13]. To describe the key idea, we briefly recall the proof idea from [13].

The proof in [13] proceeds along similar lines as the high level proof idea described above. The complexity measure used in [13] is the partial derivative measure, the same as in [15].

In the work of Nisan and Wigderson [15], $|X_i| = |X_j|$ for all $i \neq j \in [d]$. That is, they defined the hard polynomial over a variable partition where each set had the same size. The lower bound proof in [13] deviates from this and chooses the set sizes carefully. All the positive sets are of the same size, i.e. $|X_i| = |X_j| = m$ for all $i \neq j \in \mathcal{P}$, and all the negative sets are of the same size, i.e. $|X_i| = |X_j| = m'$ for all $i \neq j \in \mathcal{N}$, but $m \neq m'$. In fact, m' is $m^{1-\delta}$ for a carefully chosen δ.

This measure is then analysed for set-multilinear formulas. Suppose $F = F_1 \cdot F_2 \cdot \ldots \cdot F_r$ is a specific product gate in the formula. We consider two cases. The first case is that one of the F_is has *large* degree. In this case, the argument proceeds inductively by bounding the measure for a sub-formula inside F_i with one less product-depth.

The other case is that all the F_is have *small* degrees. The careful choice of set sizes helps in bounding the measure for factors with *small* degree (say degree equal to $O(1/\delta)$). The crucial observation is that, when the degree of F_i is small, no matter how F_i uses the subsets of positive and negative variables, the matrix M_{F_i}[9] will have many fewer columns than rows or vice versa, thereby causing some *rank deficiency*. For a term F in which all factors have small degrees, we thus obtain rank deficiency from each factor.

Here, for the non-commutative homogeneous lower bound, we use a similar proof idea. The main difference is how we analyse the measure for non-commutative homogeneous formulas. In the set-multilinear case each sub-formula depended on variable sets $(X_i : i \in J)$, where $J \subseteq [d]$. Whereas here, each sub-formula depends on a subset of variable sets $(X_i : i \in J)$, such that J is an *interval*. It turns out that we can leverage this difference and quantitatively improve the lower bounds.

The large depth case. To get lower bounds for set-multilinear formulas of product-depths greater than 1, Nisan and Wigderson combined the partial derivative measure with the method of *random restrictions*. A restriction ρ is a function that sets some of the variables to field constants. It is a random restriction if this map is random. (The choice of the distribution can play a key role in the proof.) They were able to show that the *hard polynomial* (say e.g. IMM) continues to have a high measure even after being subject to a restriction, while the measure for the set-multilinear formulas drops further under the random restriction with high probability. Using this, they proved an FPT lower bound of $\text{poly}(n) \cdot \exp(\Omega(d^{1/\Delta}))$ for product-depth $\Delta > 1$.

In their proof, for the sets of variables $X_1, \ldots, X_d$, they choose a random subset $I \subseteq [d]$ and then set all the variables in $\bigcup_{j \notin I} X_j$ to constants. Having done this, the partial derivative method is applied to the set I to get the lower bound (hence, the sets $\mathcal{P}$ and $\mathcal{N}$ are themselves random in this argument).

Unfortunately, it is not very hard to see that their choice of random restriction cannot give non-FPT lower bounds. Intuitively, this is because there are at most 2^d choices for I. Thus, we can construct a formula F_I (of polynomial size) that is resilient to a given restriction, and summing these formulas together (with suitable coefficients) yields a formula F that is resilient to all possible restrictions. So we need a new idea to strengthen their lower bound.

In our argument, we also use random 'restrictions' of some kind. However, instead of setting all the variables from a set to constants, we identify variables (randomly) inside each X_i, while setting others to 0. It is easy to see that in our setting, we have up to $(f(n))^d$ many choices for

[9] We are abusing the notion a bit and using F_i to also denote the polynomial computed by the formula F_i.

the random restriction, where $f(n) \to \infty$ as $n \to \infty$. Intuitively, this may be the reason why we are able to get a non-FPT lower bound.

2 Preliminaries

We will consider the set of words on an alphabet $A \subseteq \mathbb{Z}$. Let $w = (w_1, \dots, w_d) \in A^d$. For an interval $I \subseteq [d]$, let w_I denote $\sum_{i \in I} w_i$. We define $P_w = \{i \mid w_i \geqslant 0\}$ and $N_w = \{i \mid w_i < 0\}$, i.e., the positive and negative indices of w respectively.

We say w is *balanced* if $w_{[d]} = 0$ and *k-unbiased* if $|w_{[t]}| \leqslant k$ for $t \leqslant d$.

Given w, we denote by $\overline{X}(w)$ a tuple of d sets of variables $(X(w_1), \dots, X(w_d))$ where $|X(w_i)| = 2^{|w_i|}$.

We denote by $\mathbb{F}_{\mathrm{sm}}[\mathcal{T}]$ the set of set-multilinear polynomials over the tuple of sets of variables $\mathcal{T}$. Similarly we denote by $\mathbb{F}_{\mathrm{sm}}\langle\mathcal{T}\rangle$ the set of non-commutative set-multilinear polynomials over the tuple $\mathcal{T}$ (i.e., more formally, $\mathbb{F}_{\mathrm{sm}}\langle\mathcal{T}\rangle$ is the set of non-commutative polynomials that are linear combination of monomials where each monomial is of the form $x_1 \cdots x_d$ with $x_i \in \mathcal{T}_i$).

Notice that since $\mathcal{T} = (\mathcal{T}_1, \dots, \mathcal{T}_d)$ is chosen ordered, there is a natural bijection between $\mathbb{F}_{\mathrm{sm}}[\mathcal{T}]$ and $\mathbb{F}_{\mathrm{sm}}\langle\mathcal{T}\rangle$. Indeed, let $\pi : \mathbb{F}_{\mathrm{sm}}\langle\mathcal{T}\rangle \to \mathbb{F}_{\mathrm{sm}}[\mathcal{T}]$ be the usual projection we got by forgetting the order of the monomials. Then, given any monomial $m \in \mathbb{F}_{\mathrm{sm}}[\mathcal{T}]$, we know there is a permutation σ such that $m_i \in \mathcal{T}_{\sigma^{-1}(i)}$ for all i. It implies that $\pi^{-1}(m) = m_{\sigma(1)} \dots m_{\sigma(d)}$. So, in the set-multilinear case, π is one-to-one.

2.1 The complexity measure

Let $\mathcal{M}_w^{\mathcal{P}}$ and $\mathcal{M}_w^{\mathcal{N}}$ denote the sets of the set-multilinear monomials over only the positive and only the negative variable sets. Let $f \in \mathbb{F}_{\mathrm{sm}}[\overline{X}(w)]$, we define $M_w(f)$ as the matrix of size $|\mathcal{M}_w^{\mathcal{P}}| \times |\mathcal{M}_w^{\mathcal{N}}|$, where the rows are indexed by $\mathcal{M}_w^P$ and the columns by $\mathcal{M}_w^{\mathcal{N}}$ and where the coefficient at the entry (m_1, m_2) corresponds to the coefficient of the monomial $m_1 m_2$ in f.

We associate with the space $\mathbb{F}_{\mathrm{sm}}[\overline{X}(w)]$ the standard rank-based complexity measure relrk_w (short for "relative rank") defined as follows. Let $f \in \mathbb{F}_{\mathrm{sm}}[\overline{X}(w)]$ and define

$$\mathrm{relrk}_w(f) = \frac{\mathrm{rank}(M_w(f))}{\sqrt{|\mathcal{M}_w^{\mathcal{P}}| \cdot |\mathcal{M}_w^{\mathcal{N}}|}} = \frac{\mathrm{rank}(M_w(f))}{2^{\frac{1}{2}\sum_{i \in [d]} |w_i|}} \leqslant 1.$$

We use the following properties of relrk_w. The (standard) proof can be found in [13].

Claim 5.

1. *(Imbalance) If $f \in \mathbb{F}_{sm}[\overline{X}(w)]$, then,* $\mathrm{relrk}_w(f) \leqslant 2^{-|w_{[d]}|/2}$.
2. *(Sub-additivity) Say $f, g \in \mathbb{F}_{sm}[\overline{X}(w)]$. Then* $\mathrm{relrk}_w(f+g) \leqslant \mathrm{relrk}_w(f) + \mathrm{relrk}_w(g)$.
3. *(Multiplicativity) Say (S_1, S_2) is a partition of $[d]$. Assume $f_i \in \mathbb{F}_{sm}[\overline{X}(w_{|S_i})]$ ($i \in [2]$). Then*
$$\mathrm{relrk}_w(f_1 \cdot f_2) = \mathrm{relrk}_{w_{|S_1}}(f_1) \cdot \mathrm{relrk}_{w_{|S_2}}(f_2).$$

If f is non-commutative, i.e., if $f \in \mathbb{F}_{\mathrm{sm}}\langle\overline{X}(w)\rangle$, then we define

$$\mathrm{relrk}_w(f) = \mathrm{relrk}_w(\pi(f)).$$

We can easily notice that the two first assertions of Claim 5 still hold in the non-commutative settings since π is additive. For the third point, since the multiplication is non-commutative, the partition should be ordered. So, since π is also multiplicative we can get the similar assertion:

3. (Multiplicativity) Say $w = w_1 w_2$ (i.e., w is the concatenation of w_1 and w_2). Assume $f_i \in \mathbb{F}_{\mathrm{sm}}\langle \overline{X}(w_i)\rangle$ ($i \in [2]$). Then

$$\mathrm{relrk}_w(f_1 \cdot f_2) = \mathrm{relrk}_{w_1}(f_1) \cdot \mathrm{relrk}_{w_2}(f_2).$$

2.2 Non-commutative algebraic models of computation

We recall some definitions and facts about non-commutative algebraic models of computation (see, e.g. [14]).

Fix a variable set X and an ordered variable partition $(X_1, \ldots, X_d)$.

A *non-commutative algebraic formula* F over X is a directed tree where leaves are labelled by variables and elements of the field $\mathbb{F}$, and internal nodes are labelled by $+$ and $\times$. Each gate in F computes a polynomial in the non-commutative ring $\mathbb{F}\langle X\rangle$. Gates labelled $+$ compute a linear combination of their inputs (where the coefficients of the linear combinations are labels of the corresponding incoming edges from the children) and gates labelled $\times$ compute the products of their inputs in a fixed order. The formula is said to be *homogeneous* if each gate computes a homogeneous polynomial (of some degree). More precisely, each gate in the formula is associated with an integer d_g such that

- If g is a leaf, then $d_g = 0$ or 1 depending on whether the leaf is labelled by a constant from $\mathbb{F}$ or a variable from X.
- If g is a $+$ gate with children $g_1, \ldots, g_r$, then $d_g = d_{g_1} = \cdots = d_{g_r}$.
- If g is a $\times$ gate with children $g_1, \ldots, g_r$, then $d_g = d_{g_1} + \cdots + d_{g_r}$.

Moreover, the formula is *ordered set-multilinear* w.r.t. the ordered partition $(X_1, \ldots, X_d)$ if for each gate g of the formula, there is an interval $I_g = \{i, i+1, \ldots, i+t-1\} \subseteq [d]$ such that the polynomial computed by g lies in the space $\mathbb{F}_{\mathrm{sm}}\langle (X_i, \ldots, X_{i+t-1})\rangle$. More precisely, we have for each gate g an interval $I_g \subseteq [d]$ such that the following hold.

- If g is a leaf, then $I_g = \varnothing$ if g is labelled by a constant from $\mathbb{F}$ and $I_g = \{i\}$ if g is labelled by a variable from X_i.
- If g is a $+$ gate with children $g_1, \ldots, g_r$, then $I_g = I_{g_1} = \cdots = I_{g_r}$.
- If g is a $\times$ gate with children $g_1, \ldots, g_r$ (multiplied in this order), then I_g is a disjoint union of $I_{g_1}, \ldots, I_{g_r}$ with $\max I_{g_1} < \max I_{g_2} < \cdots < \max I_{g_r}$.

It is easy to see that if F is an ordered set-multilinear formula, then each gate g computes a homogeneous polynomial of degree $|I_g|$ and hence any non-commutative ordered set-multilinear formula is in particular homogeneous. The following lemma says that the two models are essentially equivalent.

Lemma 6. *Assume $P \in \mathbb{F}_{sm}\langle (X_1, \ldots, X_d)\rangle$ has a non-commutative homogeneous formula F of size s and product-depth at most Δ (for some $\Delta \geqslant 1$). Then P also has a non-commutative ordered set-multilinear formula of size at most s and product-depth at most Δ.*

Proof. Let F be a non-commutative homogeneous formula for P. We recall that for such a formula, each node g is associated with a degree d_g. We start by labelling the nodes g of F by intervals $I_g \subseteq [d]$ such that $|I_g| = d_g$. We do it inductively (starting by the root):

- the output node is labelled by $[d]$,
- if g is a $+$ gate with children $g_1, \ldots, g_r$, then we choose $I_{g_1} = I_{g_2} = \ldots = I_{g_r} = I_g$,

- if g is a $\times$ gate with children $g_1, \ldots, g_r$ (in this order), then we have $I_g = [a, a + d_g - 1]$ for some a. We choose for $1 \leqslant j \leqslant r$, $I_{g_j} = [a + \sum_{j'<j} d_{g_{j'}}, a + \sum_{j' \leqslant j} d_{g_{j'}} - 1]$ (notice they actually form an ordered partition of I_g).

From this labelled form of F, we define our new formula F' by modifying some of its degree-1 leaves. More precisely, if a degree-1 leaf g of F computing a variable from X_i is labelled by the interval $\{i\}$, we let it unchanged. If it is labelled by another interval, we change this leaf to 0.

To any gate g of F, we associate its corresponding gate g' in F'. We can see now by induction on the formula that the polynomial computed by g' equals the projection of the polynomial computed by g on $\mathbb{F}_{\mathrm{sm}}\langle X_{I_g}\rangle$. Indeed,

- if g is a leaf of F which computes a variable x from X_i, then its projection on $\mathbb{F}_{\mathrm{sm}}\langle X_{I_g}\rangle$ is x if $I_g = \{i\}$ and 0 otherwise,
- if g is a + gate with children $g_1, \ldots, g_r$, then the projection of g is the sum of the projections of the g_j,
- if g is a $\times$ gate with children $g_1, \ldots, g_r$, then the projection of g along the interval I_g equals the ordered product of the projections of g_j along the intervals I_{g_j}.

□

Given the above lemma, we will employ the terminology 'non-commutative homogeneous formula' or 'non-commutative set-multilinear formula' or 'non-commutative ordered set-multilinear formula' interchangeably.

A non-commutative layered *Algebraic Branching Program* [14] (ABP) A is a directed acyclic graph with layers labelled $0, \ldots, d$, where the first and last layer have a single source vertex s and sink vertex t each, all edges go from a layer labelled i to a layer labelled $i+1$ (for $0 \leqslant i < d$), and each such edge is labelled with a variable[10] in X_{i+1}. The polynomial $P_A \in \mathbb{F}_{\mathrm{sm}}\langle (X_1, \ldots, X_d)\rangle$ computed by the ABP is the sum, over all the paths ρ going from s to t, of the product of the edge labels of the edges of ρ (in increasing order of the source layer). A commutative set-multilinear ABP is defined in the same way, except that the polynomial P_A is interpreted as an element of $\mathbb{F}_{\mathrm{sm}}[(X_1, \ldots, X_d)]$.[11] The *width* of the ABP A is the maximum number of vertices in any layer.

The following fact is standard and easy to show.

Fact 7. *Given a non-commutative layered ABP A as above with width at most n, there is a set-multilinear 'reduction' from this polynomial to the Iterated Matrix Multiplication polynomial in the following sense. Fix the polynomial* $\mathrm{IMM}_{n,d}$ *defined on variable sets $Y_1, \ldots, Y_d$ with $|Y_1| = |Y_d| = n$ and $|Y_i| = n^2$ for $1 < i < d$. Then, for each $i \in [d]$ there are maps L_i mapping Y_i to $X_i \cup \{0\}$ such that applying this substitution to* $\mathrm{IMM}_{n,d}$ *yields the polynomial P_A.*

An analogous fact also holds for commutative set-multilinear ABPs.

2.3 ABPs with large relative rank

We note that for every w which does not have too much bias, there is a polynomial $P_w \in \mathbb{F}_{\mathrm{sm}}\langle \overline{X}(w)\rangle$ that has large rank w.r.t. w and is a simple projection of a small instance of the Iterated Matrix Multiplication polynomial.

[10] One can also define the labels to be homogeneous linear polynomial in X_{i+1}, but this more restrictive definition is sufficient in our setting.

[11] There is another, more general, definition of a set-multilinear ABP due to Arvind and Raja [22]. This definition is not suitable for our purposes here.

The following was proved in the commutative set-multilinear setting in our earlier work [13]. Essentially the same proof translates to the non-commutative setting.

Theorem 8. *Let $w \in A^d$ be any word that is b-unbiased. There is an explicit polynomial $P_w \in \mathbb{F}_{sm}\langle \overline{X}(w)\rangle$ such that $\mathrm{relrk}_w(P_w) \geqslant 2^{-|w_{[d]}|/2}$ and has a non-commutative layered ABP of width at most 2^b.*

Since a non-commutative set-multilinear ABP computing some polynomial f can be seen as a commutative set-multilinear ABP computing $\pi(f)$, Theorem 8 is still true in the commutative setting. (This was already proved in [13].)

Corollary 9. *Let $w \in A^d$ be any word which is b-unbiased. If there is a set-multilinear (resp. non-commutative set-multilinear) formula computing $\mathrm{IMM}_{n,d}$ of size s where $n \geqslant 2^b$, then there is also a set-multilinear (resp. non-commutative set-multilinear) formula of size at most s computing a polynomial $P_w \in \mathbb{F}_{sm}\langle \overline{X}(w)\rangle$ such that $\mathrm{relrk}_w(P_w) \geqslant 2^{-|w_{[d]}|/2}$.*

Proof. By Theorem 8, we know that there is a non-commutative layered ABP of width at most 2^b computing a polynomial $P_w \in \mathbb{F}_{\mathrm{sm}}\langle \overline{X}(w)\rangle$ such that $\mathrm{relrk}_w(P_w) \geqslant 2^{-b/2}$. By Fact 7, the polynomial P_w can be obtained from $\mathrm{IMM}_{n,d}$ via a simple variable substitution. If $\mathrm{IMM}_{2^b,d}$ has a formula F of size s, applying this substitution to F yields a formula F' (also of size at most s) computing P_w. It is easy to check that F' is a non-commutative set-multilinear formula.

For the commutative setting, we apply the set-multilinear analog of the above argument w.r.t. the polynomial $\pi(P_w)$. □

2.4 Product lemma

We recall a well-known property of formulas. Analogs of this property have been proved in various previous settings (see, e.g. [21, Chapter 3]) but as far as we know, the statement below does not appear anywhere. We give the (fairly standard) proof.

Lemma 10 (Product lemma)**.** *Assume that F is a set-multilinear formula of degree $d \geqslant 1$ with at most s leaves. Then, we can write*

$$F = \sum_{i=1}^{s} \prod_{j=1}^{\ell} G_{i,j}$$

where $\ell \geqslant \log_2 d$, every factor $G_{i,j}$ is set-multilinear and has degree at least 1, and for each $i \in [s]$, the product is set-multilinear. Moreover, if F has product-depth at most Δ with $1 \leqslant \Delta \leqslant \ln d$, then we can choose $\ell = \Delta d^{1/\Delta} - \Delta + 1$.

Proof. First notice that if a formula is of degree at least 1 and homogeneous, we can remove all the degree-0 nodes. Indeed, such a node computes a constant γ and is linked with the remainder of the circuit by a product gate (of fan-in at least two). So we can remove this gate by multiplying another fixed entry of the product by γ by adding a fan-in one addition gate and labelling the corresponding edge by γ. So, we assume that all nodes of F have degree at least 1.

We prove the result by induction on the number of leaves. If the formula has no leaves (i.e., computes 0), then the result is trivial.

So assuming that the lemma is proved for formulas of at most s leaves, let us prove it for a formula F with $s+1$ leaves.

A leaf α will be called maximal if for each multiplication gate β on the path from the root ρ to α, the child of β along this path has maximal degree amongst the children of β.

Let α be a maximal leaf of F. Let $\beta_p, \ldots, \beta_1$ ($p \leqslant \Delta$) be the multiplication gates which lie (in this order) on the path from the root to α. For each β_i, let us denote by $\beta_{i,1}$ its child on the path from the root to α and by $\beta_{i,2}, \ldots, \beta_{i,r_i}$ its other children. If μ is a gate of F, we will denote by $\hat{F}_\mu$ the polynomial computed by the subformula rooted in μ. Then, we can easily proved by induction on p that:

Claim 11.

$$\hat{F} = \hat{F}_\alpha \prod_{i=1}^{p} \left(\prod_{j=2}^{r_i} \hat{F}_{\beta_{i,j}} \right) + \hat{F}_{\alpha \leftarrow 0}$$

where $F_{\alpha \leftarrow 0}$ is the subformula we get by replacing the gate α by 0. Furthermore the product is still set-multilinear.

Proof. If $p = 0$, then $\hat{F} = \hat{F}_\alpha + \hat{F}_{\alpha \leftarrow 0}$.

Let us show the result for $p+1$. Let $F = F_1 + \ldots + F_t$ where α is in F_1 and the root of F_1 is a product gate. So in the subtree rooted in $\beta_{p+1,1}$, there are p multiplication gates between the root and α. By induction hypothesis, we have

$$\hat{F} = \left(\hat{F}_\alpha \prod_{i=1}^{p} \left(\prod_{j=2}^{r_i} \hat{F}_{\beta_{i,j}} \right) + \hat{F}_{\beta_{p+1,1} \alpha \leftarrow 0} \right) \cdot \prod_{j=2}^{r_{p+1}} \hat{F}_{\beta_{p+1,j}} + \hat{F}_2 + \ldots + \hat{F}_t.$$

The result follows from

$$\hat{F}_{\alpha \leftarrow 0} = \hat{F}_{\beta_{p+1,1} \alpha \leftarrow 0} \cdot \prod_{j=2}^{r_{p+1}} \hat{F}_{\beta_{p+1,j}} + \hat{F}_2 + \ldots + \hat{F}_t.$$

□

Let us come back to the proof of Lemma 10. The formula $F_{\alpha \leftarrow 0}$ has at most s leaves. So by induction hypothesis,

$$F_{\alpha \leftarrow 0} = \sum_{i=1}^{s} \prod_{j=1}^{\ell} G_{i,j}.$$

So it is sufficient to prove that $1 + \sum_{i=1}^{p} (r_i - 1) \geqslant \ell$. The result is trivial if $p = 0$ (and so $d = 1$). So assume that $p \geqslant 1$. The maximality condition and the homogeneity of the formula ensure that for all i, $\deg(\beta_i) \leqslant r_i \deg(\beta_{i,1}) = r_i \deg(\beta_{i+1})$. In particular, $\prod_{i=1}^{p} r_i \geqslant d$. By the AM-GM inequality, we can bound the number of factors by

$$1 + \sum_{i=1}^{p} (r_i - 1) \geqslant 1 - p + p \left(\prod_{i=1}^{p} r_i \right)^{1/p} \geqslant 1 - p + p d^{1/p}.$$

Finally, the derivative of $x \mapsto x d^{1/x}$ is negative for $0 < x < \ln d$, and positive for $x > \ln d$. Hence, the number of factors is always bounded by below by $1 + (e-1) \ln d > 1 + \log_2 d$ since $(e-1) \ln 2 > 1$. Moreover, when $\Delta \leqslant \ln d$, we have a better lower bound $\Delta d^{1/\Delta} - \Delta + 1$.

□

3 Lower bounds for large depth formulas

Here we prove Theorem 1. We start with a definition and the main observation, which follows simply from the properties of relrk from an earlier section.

Definition 12 (Bias of a word w.r.t. a partition)**.** *Let $\mathcal{S} = (S_1, \ldots, S_\ell)$ be an ordered partition of $[d]$ (each $S_i \subseteq [d]$ is non-empty). We assume that the S_is are ordered with respect to their maximal elements (i.e., $i < j \implies \max(S_i) < \max(S_j)$).*

Let $w \in \mathbb{Z}^d$ be arbitrary. Given a partition $\mathcal{S} = (S_1, S_2, \ldots, S_\ell)$ of $[d]$, we define the $\mathcal{S}$-bias of w — $\mathrm{bias}(\mathcal{S}, w)$ — *to be the quantity $\sum_{j\in[\ell]} |w_{S_j}|$ where $w_{S_j} = \sum_{i\in S_j} w_i$.*

Note that unlike the unbiased notion which we introduced earlier, the values of $\mathrm{bias}(\mathcal{S}, w)$ do not correspond to sums of w_is over some intervals, but over the parts of $\mathcal{S}$.

Lemma 13. *Let $w \in \mathbb{Z}^d$ be arbitrary. Assume $\mathcal{S} = (S_1, \ldots, S_\ell)$ is an ordered partition of $[d]$ such that* $\mathrm{bias}(\mathcal{S}, w) \geqslant r$*. Then, for* any *choice of polynomials $f_j \in \mathbb{F}_{sm}[\overline{X}(w_{|S_j})]$ ($j \in [\ell]$), we have*

$$\mathrm{relrk}_w(f_1 \cdot f_2 \cdots f_\ell) \leqslant 2^{-r/2}.$$

Proof. We know that

$$\mathrm{relrk}_w(f_1 \cdots f_r) = \prod_{j=1}^{\ell} \mathrm{relrk}_{w_{|S_j}}(f_j) \leqslant \prod_{j=1}^{\ell} 2^{-|w_{S_j}|/2} \leqslant 2^{-r/2}$$

where the equality follows from the multiplicativity of relrk and the first inequality follows from the imbalance bound (Claim 5). □

To find w that has high bias w.r.t. a given ordered interval partition, we use a random restriction idea.

Lemma 14 (Random restrictions induce high bias)**.** *Let k be any positive integer. There is a probability distribution $\mathcal{D}$ on k-unbiased $w \in \mathbb{Z}^d$ such that for any partition $\mathcal{S} = (S_1, \ldots, S_\ell)$ of $[d]$ and any $\varepsilon \geqslant 1/k$, we have*

$$\Pr_{w\sim\mathcal{D}}[\mathrm{bias}(\mathcal{S}, w) \leqslant \varepsilon k\ell] \leqslant (10\varepsilon)^{\ell/2}.$$

Proof. We first define the probability distribution $\mathcal{D}$. Choose a function $u : \{0, \ldots, d\} \to \{-k, -k+1, \ldots, k-1, k\}$ by setting $u(0) = 0$ and choosing $u(i)$ independently and uniformly at random from $\{-k, \ldots, k\}$. Now, we fix $w \in \mathbb{Z}^d$ so that $w_i = u(i) - u(i-1)$ for each $i \in [d]$. Note that for any interval, we have $w_{[t]} = u_t - u_0 = u_t$ (with $t \leqslant d$) and hence in particular w is k-unbiased.

Note the following property of w, which will be important in the sequel. Given any $i \in [d]$ and conditioned on $u(0), \ldots, u(i-1)$, the random variable w_i is uniformly distributed in some interval of length $2k+1$.

For each $j \in [\ell]$, let $\mathcal{E}_j$ denote the event that $|w_{S_j}| \leqslant 2\varepsilon k$. Note that we have

$$\begin{aligned}
\Pr_w[\mathrm{bias}(\mathcal{S}, w) \leqslant \varepsilon k\ell] &\leqslant \Pr_w[\exists B \subseteq [\ell] : |B| = \ell/2, \bigwedge_{j\in B} \mathcal{E}_j] \\
&\leqslant \sum_{B\subseteq[\ell]:|B|=\ell/2} \Pr_w[\bigwedge_{j\in B} \mathcal{E}_j] \\
&\leqslant 2^\ell \cdot \max_{B\subseteq[\ell]:|B|=\ell/2} \Pr_w[\bigwedge_{j\in B} \mathcal{E}_j].
\end{aligned}$$

So to prove the lemma it suffices to prove that for any $B \subseteq [\ell]$ of size at least $\ell/2$, we have

$$\Pr_w[\bigwedge_{j\in B} \mathcal{E}_j] \leqslant (5\varepsilon/2)^{\ell/2}. \tag{1}$$

We show that (1) follows easily using a conditioning argument. Fix some $B \subseteq [\ell]$ such that $B = \{j_1 < \cdots < j_t\}$. We have

$$\Pr_w[\bigwedge_{j\in B} \mathcal{E}_j] = \Pr_w[\mathcal{E}_{j_1} \wedge \cdots \wedge \mathcal{E}_{j_t}] = \prod_{p=1}^{t} \Pr_w[\mathcal{E}_{j_p} \mid \mathcal{E}_{j_1} \wedge \cdots \wedge \mathcal{E}_{j_{p-1}}]. \tag{2}$$

Note that for any $p \in [t]$, the event $\mathcal{E}_{j_1} \wedge \cdots \wedge \mathcal{E}_{j_{p-1}}$ depends only on $u(0), \ldots, u(q')$ where $q' = \max S_1 \cup \cdots \cup S_{j_{p-1}}$. Let $q = \max S_{j_p}$ and note that the order chosen of the parts of $\mathcal{S}$ implies $q > q'$. So let us condition on any choice of $u(0), \ldots, u(q-1)$ such that $\mathcal{E}_{j_1} \wedge \cdots \wedge \mathcal{E}_{j_{p-1}}$ holds. Now, conditioned on $u(0), \ldots, u(q-1)$ we have $w_{S_{j_p}} = w_q + \theta$ where $\theta \in \mathbb{R}$ is fixed. As noted above, conditioned on $u(0), \ldots, u(q-1)$, $w(q)$ is still uniformly distributed over an interval of length $2k+1$. In particular, the probability that $|w_{S_{j_p}}| = |w_q + \theta| \leqslant 2\varepsilon k$ is at most

$$\frac{4\varepsilon k + 1}{2k+1} \leqslant \frac{5\varepsilon k}{2k+1} \leqslant 5\varepsilon/2$$

where for the first inequality we used the fact that $\varepsilon \geqslant 1/k$. As the above holds for any conditioning of $u(0), \ldots, u(q-1)$, it implies that

$$\Pr_w[\mathcal{E}_{j_p} \mid \mathcal{E}_{j_1} \wedge \cdots \wedge \mathcal{E}_{j_{p-1}}] \leqslant 5\varepsilon/2$$

for any $p \in [t]$. By (2), we have

$$\Pr_w[\bigwedge_{j\in B} \mathcal{E}_j] \leqslant (5\varepsilon/2)^t$$

for any $B \subseteq [\ell]$ of size t. This implies (1) and thus finishes the proof of the lemma. □

We are now ready to prove the main theorem.

Proof of Theorem 1. We assume without loss of generality that n is a power of 2 and that $k = \log_2 n$. Assume $\mathrm{IMM}_{n,d}$ has a set-multilinear formula F of size at most s. We assume that the input variables of $\mathrm{IMM}_{n,d}$ are partitioned into $(X_1, \ldots, X_d)$ where X_i is the set of variables in the ith matrix.

Using the Product Lemma (Lemma 10), we have

$$\mathrm{IMM}_{n,d} = \sum_{i=1}^{s} \prod_{j=1}^{\ell} F_{i,j} \tag{3}$$

where $\ell \geqslant \log d$ and for each $i \in [s]$, there is a partition $\mathcal{S}_i = (S_{i,1}, \ldots, S_{i,\ell})$ such that $F_{i,j}$ is a set-multilinear polynomial in the variables $(X_p : p \in S_{i,j})$. If F has product-depth $\Delta \leqslant \ln d$, then we may further assume that $\ell = \Delta d^{1/\Delta} - \Delta + 1$. In both cases, if $\ell \leqslant \frac{2\log n}{\log\log n}$ then the result is trivial, so we assume this is not the case.

We will show that for any $\varepsilon \geqslant 1/k$, we have

$$s \geqslant \min\{(1/\varepsilon)^{\Omega(\ell)}, 2^{\Omega(\varepsilon k \ell)}\}. \tag{4}$$

Given the above bound, we can set $\varepsilon = (\log k/k) = \Theta(\log\log n/\log n)$ to finish the proof. It therefore suffices to prove (4).

By Lemma 14, there is a probability distribution over k-unbiased words $w \in \mathbb{Z}^d$ such that for any $\varepsilon \geqslant 1/k$ we have

$$\Pr_w[\exists i \in [s] : \mathrm{bias}(\mathcal{S}_i, w) \leqslant \varepsilon k \ell] \leqslant (10\varepsilon)^{\ell/2} \cdot s$$

where the inequality uses a union bound.

If $s \geqslant (1/10\varepsilon)^{\ell/2}$, then the inequality (4) holds trivially and we are done. So we assume $s < (1/10\varepsilon)^{\ell/2}$. In particular, we see that there is a w such that $\mathrm{bias}(\mathcal{S}_i, w) > \varepsilon k\ell$ for each $i \in [s]$ and fix such a w for the rest of the proof.

Since w is k-unbiased, we know by Corollary 9 that there is a polynomial P_w which is a set-multilinear restriction of $\mathrm{IMM}_{2^k,d} = \mathrm{IMM}_{n,d}$ such that $\mathrm{relrk}_w(P_w) \geqslant 2^{-k/2}$. Thus, by applying this linear substitution to both sides of (3) we get

$$P_w = \sum_{i=1}^{s} \prod_{j=1}^{\ell} P_{i,j}$$

where $P_{i,j}$ is the result of applying the linear substitutions to all the variables of $F_{i,j}$. Note in particular that $P_{i,j}$ is a set-multilinear polynomial in just the variables of $\overline{X}(w_{|S_{i,j}})$. Hence, by Lemma 13, we have for each $i \in [s]$,

$$\mathrm{relrk}_w\left(\prod_{j=1}^{\ell} P_{i,j}\right) \leqslant 2^{-\varepsilon k\ell/2}.$$

On the other hand, by the sub-additivity of relrk we have

$$2^{-k/2} \leqslant \mathrm{relrk}_w(P_w) \leqslant \sum_{i=1}^{s} \mathrm{relrk}_w\left(\prod_{j=1}^{\ell} P_{i,j}\right) \leqslant s \cdot 2^{-\varepsilon k\ell/2}.$$

Using the fact that $\ell \log\log n \geqslant 2 \log n$ and $\varepsilon = (\log k)/k$, this implies that $s \geqslant 2^{\Omega(\varepsilon k\ell)}$ implying (3) and finishing the proof. □

4 Non-commutative homogeneous lower bound

In this section, we prove Theorem 4, which yields a stronger lower bound for non-commutative homogeneous formulas of small depth. In view of Lemma 6, it suffices to prove the lower bound for non-commutative ordered set-multilinear formulas.

Proposition 15. *Let $d, k \geqslant 1$ and any positive integer $\Delta \leqslant (\log_2 d)/3$. There is a $w^\Delta \in \mathbb{Z}^d$ that is balanced and $k2^\Delta$-unbiased such that for any non-commutative ordered set-multilinear formula F of product-depth Δ over $\overline{X}(w^\Delta)$ of size at most s, we have*

$$\mathrm{relrk}_w(F) \leqslant s \cdot \exp(-kd^{1/\Delta}/10).$$

Corollary 16. *Let $n, d, \Delta \in \mathbb{N}\backslash\{0\}$. Any non-commutative ordered set-multilinear formula F of product-depth Δ computing $\mathrm{IMM}_{n,d}$ has size at least $n^{\Omega\left(d^{1/\Delta}/2^\Delta\right)}$.*

Proof. We assume that $\Delta \leqslant \log d/3$ since otherwise the result is trivial.

We now split the analysis into two cases. If $2^\Delta \geqslant \log n$, then we need to argue a lower bound of $\exp(\Omega(d^{1/\Delta}))$. For this, we appeal to a result of Nisan and Wigderson [15] which yields such a lower bound for *commutative* set-multilinear formulas of product-depth Δ. This also implies a lower bound for the non-commutative case, as we can just treat any non-commutative set-multilinear formula for $\mathrm{IMM}_{n,d}$ as a commutative formula for the same polynomial. Hence, we are done.

Now assume that $2^\Delta < \log n$. Fix integer $k = \lfloor \log(n^{1/2^\Delta}) \rfloor \geqslant 1$ such that there is a balanced word w as guaranteed by Proposition 15 that is $\lfloor \log n \rfloor$-unbiased. By Corollary 9, if $\mathrm{IMM}_{n,d}$ has a set-multilinear formula F of size s and product-depth Δ, then so does some polynomial P_w such that $\mathrm{relrk}_w(P_w) = 1$. By Proposition 15, we must then have $s \geqslant \exp(\Omega(kd^{1/\Delta})) = n^{\Omega(d^{1/\Delta}/2^\Delta)}$. This finishes the proof. □

Our previous result [13] proves a weaker bound than the one above for *commutative* set-multilinear formulas under the additional assumption that $d = O(\log n)$. This result is incomparable to that one, as the model is weaker but the quantitative bounds obtained are stronger.

Proof of Proposition 15. Let us begin by defining w^Δ. Let r to be the largest odd number such that $r^\Delta - 1 \leqslant d$. We work with a word w that is as defined below on the first $d_\Delta = r^\Delta - 1$ coordinates and 0 on all other coordinates. In particular we have $r \geqslant (d+1)^{1/\Delta} - 2 \geqslant d^{1/\Delta}/2 \geqslant 3$ where the last two inequalities uses $\Delta \leqslant (\log d)/3$. More generally, let $d_i = r^i - 1$ for $i \leqslant \Delta$.

Define, for each $i \leqslant \Delta$, a word $w^i \in \mathbb{Z}^{d_i}$ as follows.

$$\begin{aligned} w^1 &= (k, -k, k, -k, \cdots, k, -k) \\ w^{i+1} &= (w^i, (B_i + k), w^i, -(B_i + k), \cdots, -(B_i + k), w^i) \end{aligned} \tag{5}$$

where $B_i := \max_I |w^i_I|$ where I ranges over intervals contained in $[d_i]$ that begin at 1 or end at d_i (i.e. the intervals that measure how unbiased w^i is). More explicitly, w_1 is the word of size $d_1 = r - 1$ consisting of alternations of the values k and $-k$ (notice that d_1 is even). The construction of w^{i+1} contains exactly r copies of w^i. Between them, there are $r - 1$ locations where we again alternate between $(B_i + k)$ and $-(B_i + k)$.

It can be checked by induction on i that the following hold.

(P1) Each w^i is balanced: the sum of its entries is 0.

(P2) For every interval $I \subseteq [d_i]$ that begins at 1, $w^i_I \geqslant 0$ and for every interval $I \subseteq [d_i]$ that ends at d_i, $w^i_I \leqslant 0$.

(P3) If $i < \Delta$, then $B_{i+1} \leqslant 2B_i + k$.

The last condition implies that $B_i \leqslant k(2^i - 1) \leqslant k2^i$ for each $i \in [\Delta]$. In particular, the word w^Δ is balanced and $k2^\Delta$-unbiased as claimed.

We call the positions of w^{i+1} that contain $(B_i + k)$ or $-(B_i + k)$ the *extremal* positions of w^{i+1}. We have set things up so that w^{i+1} has exactly $r - 1$ many extremal positions.

We prove the following stronger claim by induction on the product-depth. For $i \leqslant \Delta$, let F be any (non-commutative ordered set-multilinear) formula of product-depth i over a $\overline{X}(W)$ where W is a (contiguous) subword of w^Δ that contains w^i. Then

$$\mathrm{relrk}_W(F) \leqslant s \cdot \exp(-kr/5)$$

where s denotes the size of F. This claim finishes the proof since $r \geqslant d^{1/\Delta}/2$.

The base case of the induction corresponds to product-depth 1. Let F be a formula over $\overline{X}(W)$ where W contains w^1 as a subword. We have $F = F_1 + \cdots + F_s$ where F_is are products of linear functions. Note that by Claim 5, any linear function L_j over the variable set $X(W_j)$ satisfies $\mathrm{relrk}_{W_j}(L_j) \leqslant 2^{-|W_j|/2}$. In particular, each product of linear functions F_i satisfies $\mathrm{relrk}_W(F_i) \leqslant 2^{-\sum_j |W_j|/2}$. Hence, by the subadditivity of $\mathrm{relrk}_W(\cdot)$, we have

$$\begin{aligned} \mathrm{relrk}_W(F) &\leqslant \sum_{i=1}^{s} \mathrm{relrk}_W(F_i) \\ &\leqslant s \cdot 2^{-\sum_j |W_j|/2} \\ &\leqslant s \cdot 2^{-\sum_j |w^1_j|/2} \\ &\leqslant s \cdot \exp(-kr/5) \end{aligned}$$

where for the second-last inequality we used the fact that w^1 is a subword of W and for the last one, the fact that $r \geqslant 3$.

Now for the inductive case. Assume the above is already proved for depth $i < \Delta$ and consider depth $i+1$. Again assume that W contains w^{i+1} and F has product-depth $i+1$. We have $F = F_1 + \cdots + F_s$ where each F_i has a product gate as output gate. By subadditivity of relrk_W it suffices to show that for each $i \in [s]$, we have

$$\mathrm{relrk}_W(F_i) \leqslant s_i \cdot \exp(-kr/5), \tag{6}$$

where s_i is the size of the subformula F_i.

Fix any F_i. We have $F_i = G_1 \cdots G_\ell$ which corresponds to splitting the word W into ℓ disjoint subwords $W^1, \ldots, W^\ell$. Let $s_{i,j}$ denote the size of G_j. We consider two cases.

1. **There is a $j \in [\ell]$ such that W^j contains a copy of w^i**: In this case, we can bound $\mathrm{relrk}_W(F_i)$ by

$$\mathrm{relrk}_W(F_i) \leqslant \mathrm{relrk}_{W^j}(G_j) \leqslant s_{i,j} \cdot \exp(-kr/5) \leqslant s_i \cdot \exp(-kr/5).$$

2. **There is no such $j \in [\ell]$**: In this case, the extremal positions of w^{i+1} are in different words (if two extremal positions belonged to the same word W^j, then W^j would contain a copy of w^i, which is assumed to be false). Let $W^{j_1}, \ldots, W^{j_{r-1}}$ be the words that contain the extremal positions.

   By the construction of w^{i+1} in (5), each such word W^{j_p} ($p \in [r-1]$) is a (possibly empty) partial suffix u of a copy of w^i, followed by the extremal position, which is then followed by a (possibly empty) partial prefix v of w^i. By our choice of B_i, it follows that the sum of entries of W^{j_p} is at least k in absolute value for each $p \in [r-1]$. To see this, note that if the entry in the extremal position is *positive*, then by (P2) above, the sum of the entries of v only increase this value, while the sum of the entries of u can only reduce this value by B_i, hence implying that the overall sum is at least k. A similar argument shows that if the extremal position has a negative entry, the sum of the entries of W^{j_p} is at most $-k$ and hence at least k in absolute value.

   Hence, by Claim 5, we have $\mathrm{relrk}_{W^{j_p}}(G_{j_p}) \leqslant 2^{-k/2}$ for each $p \in [r-1]$. We can thus bound $\mathrm{relrk}_W(F_i)$ by

$$\mathrm{relrk}_W(F_i) \leqslant 2^{-k(r-1)/2} \leqslant \exp(-kr/5) \leqslant s_i \exp(-kr/5).$$

We have thus proved (6) which completes the induction.

□

Finally, we can even get a depth hierarchy result. We notice that there are polynomials P_w^Δ set-multilinear over $\overline{X}(w^\Delta)$ which are computable by non-commutative ordered set-multilinear polynomial-sized formulas of product-depth $\Delta + 1$ where the words w are those chosen in the proof of Proposition 15. Intuitively, these polynomials are constructed from nested inner products according to w. Indeed, let us define

$$P_w^1\left(\overline{X}_{[a,a+r-2]}\right) = \prod_{u=1}^{(r-1)/2} \sum_{v=1}^{2^k} x_{a+2u-2,v} x_{a+2u-1,v},$$

and $P_w^{i+1}\left(\overline{X}_{[a,a+r^{i+1}-2]}\right) =$

$$P_w^i\left(\overline{X}_{[a,a+r^i-2]}\right) \cdot \prod_{u=1}^{(r-1)/2} \sum_{v=1}^{2^{B_i+k}} \prod_{j=1}^{2} x_{a+(2u+j-2)r^i-1,v} P_w^i\left(\overline{X}_{i,a,u,j}\right)$$

where $\overline{X}_{[a,b]}$ corresponds to the sets of variables $\bigcup_{i\in[a,b]} X_i$ and

$$\overline{X}_{i,a,u,j} = \overline{X}_{[a+(2u+j-2)r^i, a+(2u+j-1)r^i-2]}.$$

otice that P^i_w is always associated with an interval of variables of length $r^i - 1$. Finally, each set X_i has $2^{B_i+k} \leqslant 2^{k2^i}$ variables, so the polynomials P^Δ_w depend on at most $N = d2^{k2^\Delta}$ variables.

It is clear by definition that P^Δ_w is computed by a non-commutative ordered set-multilinear circuit of product-depth $\Delta + 1$ and size at most $\sum_{i=1}^{\Delta} r2^{B_i+k}(r^{\Delta-i}) \leqslant 2\Delta d2^{k2^\Delta} \leqslant O(\Delta \cdot N)$. Moreover, the inner product structure ensures that $\mathrm{relrk}(P^i_w) = 1$ for all i. Consequently, combined with Proposition 15, we get

Corollary 17. *Let Δ be a positive constant and N, d be growing parameters. There exist non-commutative homogeneous N-variate polynomials of degree d which are computed by non-commutative ordered set-multilinear formulas of product-depth $\Delta + 1$ and size $O(N)$, but such that any such formula of product-depth Δ has size at least*

$$s \geqslant N^{\Omega(d^{1/\Delta})}.$$

Finally, we can notice that the commutative version of P^Δ_w (i.e., $\pi(P^\Delta_w)$) is of the form

$$\prod_{u=1}^{d/2} \sum_{v=1}^{s_u} x_{\tau_2(u),v} x_{\tau_1(u),v}$$

where $s_u \leqslant \max_i(2^{B_i+k}) \leqslant N$ and τ_1, τ_2 are two one-to-one functions from $[1, d/2]$ to $[1, d]$ of disjoint images.

Consequently, it also implies a separation from constant-depth commutative formulas and constant-depth non-commutative formulas. But in fact it was already known. Nisan [14] proved that the polynomial

$$\mathrm{Pal}_n = \sum_{w\in\{0,1\}^n} X_{1,w_1} X_{2,w_2} \cdots X_{n,w_n} Y_{n,w_n} Y_{n-1,w_{n-1}} \cdots Y_{1,w_1}$$

requires non-commutative formulas of exponential size (even more so when the depth is constant). And clearly the commutative version of Pal_n can be computed by a small depth-4 formula

$$(X_{1,1}Y_{1,1} + X_{1,2}Y_{1,2})\,(X_{2,1}Y_{2,1} + X_{2,2}Y_{2,2}) \cdots (X_{n,1}Y_{n,1} + X_{n,2}Y_{n,2})\,.$$

References

[1] Richard P. Brent. The parallel evaluation of general arithmetic expressions. *Journal of the ACM*, 21(2):201–206, April 1974.

[2] Peter Bürgisser, Michael Clausen, and Mohammad Amin Shokrollahi. *Algebraic complexity theory*, volume 315 of *Grundlehren der mathematischen Wissenschaften*. Springer, 1997.

[3] Prerona Chatterjee. Separating abps and some structured formulas in the non-commutative setting. *arXiv preprint arXiv:2103.00864*, 2021.

[4] Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Springer Publishing Company, Incorporated, 2013.

[5] Zeev Dvir, Guillaume Malod, Sylvain Perifel, and Amir Yehudayoff. Separating multilinear branching programs and formulas. In *proceedings of Symposium on Theory of Computing (STOC)*, pages 615–624, 2012.

[6] Ankit Gupta, Pritish Kamath, Neeraj Kayal, and Ramprasad Saptharishi. Approaching the chasm at depth four. *J. ACM*, 61(6):33:1–33:16, December 2014.

[7] Pavel Hrubeš and Amir Yehudayoff. Homogeneous formulas and symmetric polynomials. *Computational Complexity*, 20(3):559–578, 2011.

[8] K. Kalorkoti. A lower bound for the formula size of rational functions. In *Proceedings of the 9th Colloquium on Automata, Languages and Programming*, page 330–338, Berlin, Heidelberg, 1982. Springer-Verlag.

[9] Neeraj Kayal. An exponential lower bound for the sum of powers of bounded degree polynomials. *Electron. Colloquium Comput. Complex.*, 19:81, 2012.

[10] Neeraj Kayal, Nutan Limaye, Chandan Saha, and Srikanth Srinivasan. An exponential lower bound for homogeneous depth four arithmetic formulas. *SIAM Journal on Computing*, 46(1):307–335, 2017.

[11] Neeraj Kayal, Chandan Saha, and Sébastien Tavenas. An Almost Cubic Lower Bound for Depth Three Arithmetic Circuits. In *43rd International Colloquium on Automata, Languages, and Programming (ICALP 2016)*, volume 55 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 33:1–33:15, 2016.

[12] Guillaume Lagarde, Nutan Limaye, and Srikanth Srinivasan. Lower bounds and pit for non-commutative arithmetic circuits with restricted parse trees. *Computational Complexity*, 28(3):471–542, 2019.

[13] Nutan Limaye, Srikanth Srinivasan, and Sébastien Tavenas. Superpolynomial lower bounds against low-depth algebraic circuits. *Electron. Colloquium Comput. Complex.*, 81, 2021.

[14] Noam Nisan. Lower bounds for non-commutative computation. In *Proceedings of the twenty-third annual ACM symposium on Theory of computing*, pages 410–418, 1991.

[15] Noam Nisan and Avi Wigderson. Lower bounds on arithmetic circuits via partial derivatives. *Computational Complexity*, 6(3):217–234, 1997.

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

[17] Ran Raz. Multi-linear formulas for permanent and determinant are of super-polynomial size. *J. ACM*, 56(2):8:1–8:17, 2009.

[18] Ran Raz. Tensor-rank and lower bounds for arithmetic formulas. *Journal of the ACM*, 60(6):40:1–40:15, 2013.

[19] Ramprasad Saptharishi. A survey of lower bounds in arithmetic circuit complexity. Github survey, 2015.

[20] Amir Shpilka and Avi Wigderson. Depth-3 arithmetic circuits over fields of characteristic zero. *Computational Complexity*, 10(1):1–27, 2001.

[21] Amir Shpilka and Amir Yehudayoff. Arithmetic circuits: A survey of recent results and open questions. *Foundations and Trends in Theoretical Computer Science*, 5:207–388, March 2010.

[22] S. Raja V. Arvind. Some lower bound results for set-multilinear arithmetic computations. *Chicago Journal of Theoretical Computer Science*, 2016(6).