comput. complex. (2021) 30:14

https://doi.org/10.1007/s00037-021-00214-1
computational complexity

LOWER BOUNDS FOR ARITHMETIC CIRCUITS VIA THE HANKEL MATRIX

Nathanaël Fijalkow, Guillaume Lagarde, Pierre Ohlmann, and Olivier Serre

Abstract. We study the complexity of representing polynomials by arithmetic circuits in both the commutative and the non-commutative settings. Our approach goes through a precise understanding of the more restricted setting where multiplication is not associative, meaning that we distinguish $(xy)z$ from $x(yz)$.
Our first and main conceptual result is a characterization result: We show that the size of the smallest circuit computing a given non-associative polynomial is exactly the rank of a matrix constructed from the polynomial and called the Hankel matrix. This result applies to the class of all circuits in both commutative and non-commutative settings, and can be seen as an extension of the seminal result of Nisan giving a similar characterization for non-commutative algebraic branching programs.
The study of the Hankel matrix provides a unifying approach for proving lower bounds for polynomials in the (classical) associative setting. Our key technical contribution is to provide generic lower bound theorems based on analyzing and decomposing the Hankel matrix. We obtain significant improvements on lower bounds for circuits with many parse trees, in both (associative) commutative and non-commutative settings, as well as alternative proofs of recent results proving superpolynomial and exponential lower bounds for different classes of circuits as corollaries of our characterization and decomposition results.

Keywords. Arithmetic circuit complexity, Lower bounds, Parse trees, Hankel matrix

Subject classification. 68Q17

Published online: 08 October 2021 Birkhäuser

1. Introduction

The model of arithmetic circuits is the algebraic analogue of Boolean circuits: The latter computes Boolean functions and the former computes polynomials, replacing OR gates by addition and AND gates by multiplication. Computational complexity theory is concerned with understanding the expressive power of such models. A rich theory investigates the algebraic complexity classes **VP** and **VNP** introduced by Valiant (1979). A widely open problem in this area of research is to explicitly construct hard polynomials, meaning for which we can prove superpolynomial lower bounds. To this day the best general lower bounds for arithmetic circuits were given by Baur & Strassen (1983) for the polynomial $\sum_{i=1}^{n} x_i^d$, which requires $\Omega(n \log d)$ operations. The seminal paper of Nisan (1991) initiated the study of non-commutative computation: In this setting, variables do not commute, and therefore xy and yx are considered as being two distinct monomials. Non-commutative computations arise in different scenarios, the most common mathematical examples being when working with algebras of matrices, group algebras of non-commutative groups or the quaternion algebra. A second motivation for studying the non-commutative setting is that it makes it easier to prove lower bounds which can then provide powerful ideas for the commutative case. Indeed, commutativity allows a circuit to rely on cancellations and to share calculations across different gates, making them more complicated to analyze.

1.1. Nisan's characterization for ABP. The main result of Nisan (1991) is to give a characterization of the smallest ABP computing a given polynomial. As a corollary of this characterization, Nisan obtains exponential lower bounds for the non-commutative permanent against the subclass of circuits given by ABPs.

We sketch the main ideas behind Nisan's characterization, since our first contribution is to extend these ideas to the class of all non-associative circuits. An ABP is a layered graph with two distinguished vertices, a source and a target. The edges are labelled by affine functions in a given set of variables. An ABP computes a polynomial obtained by summing over all paths from the source to the target, with the value of a path being the multiplication of the affine functions along the traversed edges. Following Nisan, fix a polynomial f, and define a matrix N_f whose rows and columns are indexed by monomials: for u, v two monomials, let $N_f(u, v)$ denote the coefficient of the monomial $u \cdot v$ in f.

The beautiful and surprisingly simple characterization of Nisan states that for a homogeneous (i.e., all monomials have the same degree) non-commutative polynomial f, the size of the smallest ABP computing f is exactly the rank of N_f. The key idea is to decompose the computation arising in the ABP, say $\mathcal{C}$: To any vertex r in $\mathcal{C}$, we associate two polynomials L_r and R_r that are respectively the one computed by the ABP induced by the original source of $\mathcal{C}$ and target r and the one computed by the ABP induced by source r and the original target of $\mathcal{C}$. For a polynomial f and a monomial m, we use $f(m)$ to denote the coefficient of m in f. For u, v two monomials, we observe that the coefficient of $u \cdot v$ in f is equal to $\sum_r L_r(u)R_r(v)$, where r ranges over all vertices of $\mathcal{C}$, $L_r(u)$ is the coefficient of u in L_r, and $R_r(v)$ is the coefficient of v in R_r. We see this as a matrix equality: $N_f = \sum_r L_r \cdot R_r$, where L_r is seen as a column vector, and R_r as a row vector. By subadditivity of the rank and since the product of a column vector by a row vector is a matrix of rank at most 1, this implies that the rank of N_f is bounded by the size of the ABP, yielding the lower bound in Nisan's result.

The crucial idea of splitting the computation of a monomial into two parts had been independently developed by Fliess when studying so-called *Hankel Matrices* in Fliess (1974) to derive a very similar result in the field of *weighted automata*, which are finite state machines recognizing *words series*, i.e., functions from finite words into a field. Fliess' theorem (Fliess 1974, Th. 2.1.1) states that the size of the smallest weighted automaton recognizing a word series f is exactly the rank of the Hankel matrix of f. The key insight to relate the two results is to see a non-commutative monomial as a finite word over the alphabet whose letters are the variables. Using this correspondence, one can obtain Nisan's theorem from Fliess' theorem, observing that the Hankel matrix coincides with the matrix N_f defined by Nisan and that acyclic weighted automata correspond to ABPs. (We refer to an early technical report of this work for more details on this correspondence (Fijalkow *et al.* 2018).)

1.2. Non-associative computations. Hrubeš *et al.* (2011) drop the associativity rule and show how to define the complexity classes **VP** and **VNP** in the absence of either commutativity or associativity (or both) and prove that these definitions are sound in particular by obtaining the completeness of the permanent.

In the same way that a non-commutative monomial can be seen as a word, a non-commutative and non-associative monomial such as $(xy)(x(zy))$ can be seen as a tree, and more precisely as an ordered binary rooted tree whose leaves are labelled by variables. The starting point of our work was to exploit this connection. The work of Bozapalidis & Louscou-Bozapalidou (1983) extends Fliess' result to trees; although we do not technically rely on their results, they serve as a guide, in particular for understanding how to decompose trees.

Let us return to the key idea in Nisan's proof, which is to decompose the computation of an ABP into two parts. The way a monomial, e.g., $x_1x_2x_3\cdots x_d$, is evaluated in an ABP is very constrained, namely from left to right, or if we make the implicit non-associative structure explicit as $w = (\cdots(((x_1x_2)x_3)x_4)\cdots)x_d$. The decompositions of w into two monomials u, v are of the form $u = (\cdots((x_1x_2)x_3)\cdots)x_{i-1})$ and $v = (\cdots((\square x_i)x_{i+1})\cdots)x_d$. Here $\square$ is a new fresh variable (the *hole*) to be substituted by u. Moving to non-associative polynomials, a monomial is a tree whose leaves are labelled by variables. A *context* is a monomial over the set of variables extended with a new fresh one denoted $\square$ and occurring exactly once. For instance (see Figure 1.1) the composition of the monomial $t = z((xx)y)$ with the context $c = (xy)((z\square)y)$ is the monomial $c[t] = (xy)((z(z((xx)y)))y)$.

Let f be a non-associative (possibly commutative) polynomial f, the *Hankel matrix* H_f of f is defined as follows: the rows of H_f are indexed by contexts and the columns by monomials, and the value of $H_f(c, t)$ at row c and column t is the coefficient of the monomial $c[t]$ in f.

Extending Nisan's proof to computations in a *general circuit*, which are done along trees, we obtain a characterization in the

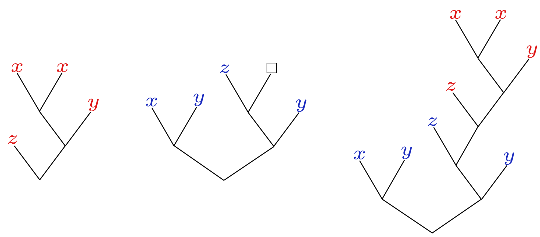


Figure 1.1: On the left-hand side the monomial t, in the middle the context c, and on the right-hand side the monomial $c[t]$.

non-associative setting (a more precise statement is given by Theorem 2.4).

THEOREM. *Let f be a non-associative homogeneous polynomial and let H_f be its Hankel matrix. Then, the size of the smallest circuit computing f is exactly rank(H_f).*

Note that this is a characterization result: the Hankel matrix exactly captures the size of the smallest circuit computing f (upper and lower bounds), exactly as in Nisan's result. Hence, understanding the rank of the Hankel matrix is equivalent to studying circuits for f. We recover and extend Nisan's characterization as a special case of our result.

1.3. Parse trees. At an intuitive level, parse trees can be used to explain in what way a circuit uses the associativity rule. Consider the case of a circuit computing the (associative) monomial $2xyz$. Since this monomial corresponds to two non-associative monomials: $(xy)z$ and $x(yz)$, the circuit may sum different computations, for instance $3(xy)z - x(yz)$, which up to associativity is $2xyz$. We say that such a circuit contains two parse trees, corresponding to the two different ways of parenthesizing xyz.

The *shape* of a non-associative monomial is the tree obtained by forgetting the variables, e.g., the shape of $(z((xy)((xx)y)))$ is $(\_\ ((\_\ \_)((\_\ \_)\ \_)))$. The parse trees of a circuit $\mathcal{C}$ are the shapes induced by computations in $\mathcal{C}$.

Many interesting classes of circuits can be defined by restricting the set of allowed parse trees, both in the commutative and the non-commutative setting.

- The simplest such class is that of Algebraic Branching Programs (ABP) (Dvir *et al.* 2012; Nisan 1991; Ramya & Rao 2018), whose only parse trees are left-combs, that is, the variables are multiplied sequentially.

- Lagarde *et al.* (2016) introduced the class of Unique Parse Tree circuits (UPT), which are circuits computing non-commutative homogeneous (but associative) polynomials such that all monomials are evaluated in the same non-associative way.

- The class of skew circuits (Allender *et al.* 1998; Limaye *et al.* 2016; Malod & Portier 2008; Toda 1992) and its extension to small non-skew depth circuits (Limaye *et al.* 2016), together with the class of unambiguous circuits (Arvind & Raja 2016) are all defined via parse tree restrictions.

- We propose in our technical developments some related restrictions called slightly balanced and slightly unbalanced circuits.

- Last but not least, the class of k-PT circuits (Arvind & Raja 2016; Lagarde *et al.* 2018; Saptharishi & Tengse 2017) is simply the class of circuits having at most k parse trees.

1.4. Contributions and outline. In this paper, we prove lower bounds for classes of circuits with parse tree restrictions, both in the commutative and non-commutative setting.

Our first and conceptually main contribution is the characterization result stated in Theorem 2.4 which gives an algebraic approach to understanding circuits in the non-associative setting. All the subsequent results in this paper are based on this approach.

Section 3.1 and Section 3.2 are devoted to the definition of parse trees and a classical tool for proving lower bounds, partial derivative matrices. We can already show at this point (in Section 3.3) how Theorem 2.4 can be specialized to give a characterization result for UPT circuits, extending Nisan's result. We note that a characterization result for UPT circuits was already known (Lagarde *et al.* 2016), we slightly improve on it. As a corollary we obtain exponential lower bounds on the size of the smallest UPT circuit computing the permanent.

Our most technical developments are discussed in Section 4. We prove generic lower bound results by further analyzing and decomposing the Hankel matrix, with the following proof scheme. We consider a polynomial f in the associative setting. Let $\mathcal{C}$ be a circuit computing f. Forgetting about associativity we can see $\mathcal{C}$ as computing a non-associative polynomial $\tilde{f}$, which projects onto f, meaning is equal to f assuming associativity. This induces a set of linear constraints: for instance if the monomial xyz has coefficient 3 in f, then we know that $\tilde{f}((xy)z) + \tilde{f}(x(yz)) = 3$. We make use of the linear constraints to derive lower bounds on the rank of the Hankel matrix $H_{\tilde{f}}$, yielding a lower bound on the size of $\mathcal{C}$.

The final section is devoted to applications of our results, where we obtain superpolynomial and exponential lower bounds for various classes. In the results mentioned below, n is the number of variables, d is the degree of the polynomial, and k the number of parse trees. We note that the lower bounds hold for any (prime) n, any d, and any field.

We obtain alternative proofs of some known lower bounds: unambiguous circuits (Arvind & Raja 2016), skew circuits (Limaye *et al.* 2016) and small non-skew depth circuits (obtaining a much shorter proof than (Limaye *et al.* 2016)).

Our novel results are:

- *Slightly unbalanced circuits.* We extend the exponential lower bound from (Limaye *et al.* 2016) on $\frac{1}{5}$-unbalanced circuits to $\left(\frac{1}{2} - \varepsilon\right)$-unbalanced circuits.

- *Slightly balanced circuits.* We derive a new exponential lower bound for ε-balanced circuits.

- *Circuits with k parse trees in the non-commutative setting.* We substantially extend the superpolynomial lower bound of (Lagarde *et al.* 2018) from $k = 2^{d^{1/3-\varepsilon}}$ to $k = 2^{d^{1-\varepsilon}}$, the total number of possible non-commutative parse trees being $2^{O(d)}$.

- *Circuits with k parse trees in the commutative setting.* We substantially extend the superpolynomial lower bound from (Arvind & Raja 2016) from $k = d^{1/2-\varepsilon}$ to $k = 2^{d^{1/3-\varepsilon}}$, and even to $k = 2^{d^{1-\varepsilon}}$, when d is polylogarithmic in n.

1.5. Related work. We argued that proving lower bounds in the non-commutative setting is easier, but this has not yet materialized since the best lower bound for general circuits in this setting is the same as in the commutative setting (by Baur and Strassen, already mentioned above). Indeed, recent impressive results suggest that this may be hard: Carmosino *et al.* (2018) (essentially) proved that a lower bound in the non-commutative setting which would be slightly stronger than superlinear can be amplified to get strong lower bounds (even exponential, in some cases) again in the non-commutative setting.

Most approaches for proving lower bounds rely on algebraic techniques and the rank of some matrix. A different and beautiful approach was investigated by Hrubeš *et al.* (2011) in the non-commutative setting through the study of the so-called *sum-of-squares problem*. Roughly speaking, the goal is to decompose $(x_1^2 + \cdots + x_k^2) \cdot (y_1^2 + \cdots + y_k^2)$ into a sum of n squared bilinear forms in the variables x_i and y_j. They show that almost any superlinear bound on n implies non-trivial lower bounds on the size of any non-commutative circuit computing the permanent.

The quest of finding lower bounds is deeply connected to another problem called polynomial identity testing (PIT) for which the goal is to decide whether a given circuit computes the formal zero polynomial. The connection was shown in Kabanets & Impagliazzo (2003), in which it is proved that providing an efficient deterministic algorithm to solve the problem implies strong lower bounds either in the arithmetic or boolean setting. PIT was widely investigated in the commutative and non-commutative settings for classes of circuits based on parse trees restrictions, see e.g., Agrawal *et al.* (2015); Arvind *et al.* (2017); Forbes *et al.* (2014); Gurjar *et al.* (2017); Raz & Shpilka (2005); Saptharishi & Tengse (2017).

2. Characterizing non-associative circuits

2.1. Basic definitions. For an integer $d \in \mathbb{N}$, we let $[d]$ denote the integer interval $\{1, \ldots, d\}$.

Polynomials. Let K be a field and let X be a set of ***variables***. Following Hrubeš *et al.* (2011) we consider that unless otherwise stated multiplication is neither commutative nor associative. We assume however that addition is commutative and associative, and that multiplication distributes over addition. A ***monomial*** is a product of variables in X and a polynomial f is a formal finite sum $\sum_i c_i m_i$ where m_i is a monomial and $c_i \in K$ is a nonzero element called the coefficient of m_i in f. We let $f(m_i)$ denote the coefficient of m_i in f, so that $f = \sum_i f(m_i) m_i$.

The ***degree*** of a monomial is defined in the usual way, i.e., $\deg(x) = 1$ when $x \in X$ and $\deg(m_1 m_2) = \deg(m_1) + \deg(m_2)$; the degree of a polynomial f is the maximal degree of a monomial in f. A polynomial is ***homogeneous*** if all its monomials have the same degree. Depending on whether we include the relations $u \cdot v = v \cdot u$ (commutativity) and $u \cdot (v \cdot w) = (u \cdot v) \cdot w$ (associativity) we obtain four classes of polynomials.

Unless otherwise specified, for a polynomial f we use n for the number of variables and d for the degree.

Trees and contexts. The ***trees*** we consider have a single root and binary branching (every internal node has exactly two children). To account for the commutative and for the non-commutative setting we use either ***unordered trees*** or ***ordered trees***, the only difference being that in the case of ordered trees we distinguish the left child from the right child. We let ***Tree*** denote the set of trees (it will be clear from the context whether they are ordered or not). The size of a tree is defined as its number of leaves.

A ***non-associative monomial*** t is a tree with leaves labelled by variables. If t is non-commutative then it is an ordered tree, and if t is commutative then it is an unordered tree. We let $\boldsymbol{Tree(X)}$ denote the set of trees whose leaves are labelled by variables in X and $\boldsymbol{Tree_i(X)}$ denote the subset of such trees with i leaves, which are monomials of degree i. Given a non-associative monomial t, we let $\mathrm{label}(t)$ be the associative monomial corresponding to the multiplication of the variables at the leaves of t. If t is non-commutative, the multiplication is done from left to right, and $\mathrm{label}(t)$ is a non-commutative monomial, that is, a word.

In this paper, we see polynomials as finitely supported mappings from monomials to K. For instance, in the non-associative setting where monomials are trees, a non-associative polynomial is a map $\mathrm{Tree}(X) \to K$. To avoid possible confusion, let us insist that the notation $f(t)$ refers to the coefficient of the monomial t in the polynomial f, not to be confused with the evaluation of f at a given point.

A (ordered or unordered) ***context*** is a tree with a distinguished leaf labelled by a special symbol called the ***hole*** and written $\square$. We let ***Context(X)*** denote the set of contexts whose leaves are labelled by variables in X. Given a context c and a tree t we construct a new tree $c[t]$ by substituting the hole of c by t. This operation is defined in both ordered and unordered settings. See Figure 1.1 for an example. It can be read in both the ordered or unordered settings.

Hankel matrices. Let f be a non-associative polynomial. The ***Hankel matrix*** H_f of f is the matrix whose rows are indexed by contexts and columns by monomials and such that the value of H_f at row c and column t is the coefficient of the monomial $c[t]$ in f. Note that H_f is an infinite matrix with finite support, so its rank is well defined. As we will be interested in computing the rank of H_f, we freely depict its rows and columns ordered arbitrarily and conveniently.

Arithmetic circuits. An (arithmetic) ***circuit*** is a directed acyclic graph such that the vertices are of three types:

- input gates: they have in-degree 0 and are labelled by variables in X,
- addition gates: they have arbitrary in-degree, an output weight in K, and a weight $w(a) \in K$ on each incoming arc a,
- multiplication gates: they have in-degree 2, and we distinguish between the left child and the right child.

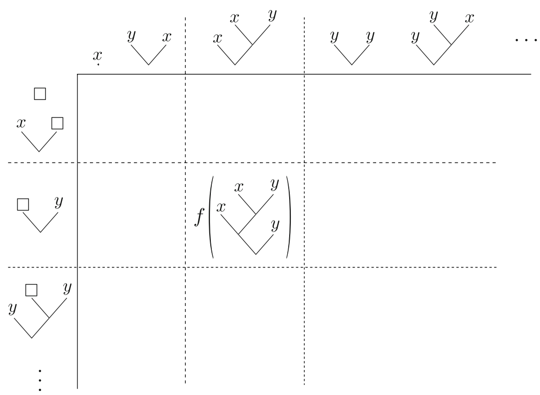


Figure 2.1: A depiction of the Hankel matrix of a non-associative polynomial f. Only one coefficient is displayed for clarity.

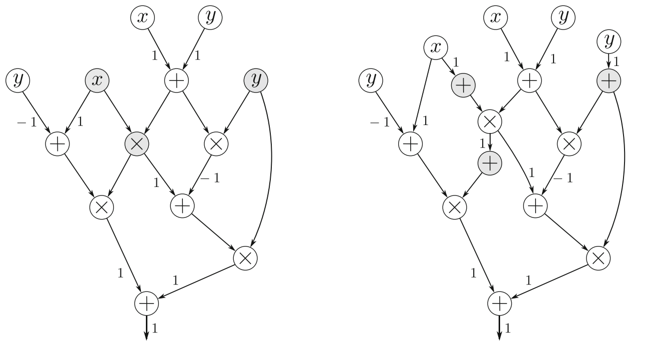
Each gate v in the circuit computes a polynomial f_v which we define by induction.

- An input gate labelled by a variable $x \in X$ computes the polynomial x.
- An addition gate v with n arcs incoming from gates $v_1, \ldots, v_n$ and with weights $\alpha_1, \ldots, \alpha_n$, computes the polynomial $\alpha_1 f_{v_1} + \cdots + \alpha_n f_{v_n}$.
- A multiplication gate with left child u and right child v computes the polynomial $f_u f_v$.

The circuit itself computes a polynomial given by the sum over all addition gates of the polynomial computed by the gate times its output weight. Note that it is slightly unusual that all addition gates contribute to the circuit; one can easily reduce to the classical case where there is a unique output addition gate by adding an extra gate.

We shall make a syntactic assumption: each arc is either coming from, or going to (but not both), an addition gate. Any circuit can be put into this form by adding addition gates, at most one per input gate and per multiplication gate (see Figure 2.2). We also ask two input gates referring to the same variable to not feed the same addition gate. We then define the size of a circuit to be its number of addition gates, which compensates this small blow up. Doing so we slightly differ from usual; however, this will allow our characterization result to be exact.

Note that the definitions we gave above do not depend on which of the four settings we consider: commutative or non-commutative, associative or non-associative.



	A	NA
C	$x^3 + x^2y - y^2x - y^3$	$x(xx) + x(xy) - y(xy) - y(yy)$
NC	$x^3 + 2x^2y - yx^2$ $- yxy - y^3$	$x(xx) + x(xy) + (xx)y - y(xx) - y(xy) - (yy)y$

Figure 2.2: The circuit on the left does not satisfy our syntactic assumption because of the edges leaving the greyed gates. However, the one on the right, obtained by adding two addition gates does satisfy the assumption. It has size 6. Both circuits compute the same polynomials in each setting, which are given in the table below, where the abbreviations A, NA, C, NC respectively stand for associative, non-associative, commutative, non-commutative. We use labelled outgoing edges to depict output weights, and omit them when the output weight is 0.

2.2. The characterization. This section aims at proving the characterization stated in Theorem 2.4 below—the Hankel matrix H_f exactly captures (upper and lower bounds) the size of the smallest circuit computing f—, extending Nisan's characterization of non-commutative ABPs to general circuits in the non-associative setting. The result holds for both commutative and non-commutative settings, the proof being the same up to cosmetic changes.

The key step to go from ABPs to general circuits is the following: the polynomial computed by an ABP is the sum over the *paths* of the underlying graph, whereas in a general circuit the sum is over *trees*. We formalize this in the next definition by introducing *runs* of a circuit. The definition is given in the non-commutative setting but easily adapts to the commutative setting as explained later in Remark 2.2.

Definition 2.1. *Let $\mathcal{C}$ be a circuit and $V_\oplus$ denote its set of addition gates. Let $t \in \mathit{Tree}(X)$ be a monomial. A **run of $\mathcal{C}$ over t** is a map ρ from nodes of t to $V_\oplus$ such that*

(i) A leaf of t with label $x \in X$ is mapped to a gate with a nonzero edge incoming from an input gate labelled by x.

(ii) If n is a node of t with left child n_1 and right child n_2, then $\rho(n)$ has a nonzero edge incoming from a multiplication gate with left child $\rho(n_1)$ and right child $\rho(n_2)$.

*The **value** $\mathrm{val}(\rho)$ of ρ is a nonzero element in K defined as the product of the weights of the edges mentioned in items (i) and (ii) together with the output weight of $\rho(r)$, r being the root of t.*

We write by a slight abuse of notation $\rho : t \to V_\oplus$ for runs of $\mathcal{C}$ over t.

Figure 2.3 depicts a run in the circuit from Figure 2.2.

Remark 2.2. *In the commutative setting we simply replace item (ii) by: "if n is a node of t with children n_1, n_2, then $\rho(n)$ has a nonzero edge incoming from a multiplication gate with children $\rho(n_1), \rho(n_2)$".*

A run of $\mathcal{C}$ over a monomial t additively contributes to the coefficient of t in the polynomial computed by $\mathcal{C}$, leading to the following straighforward lemma.

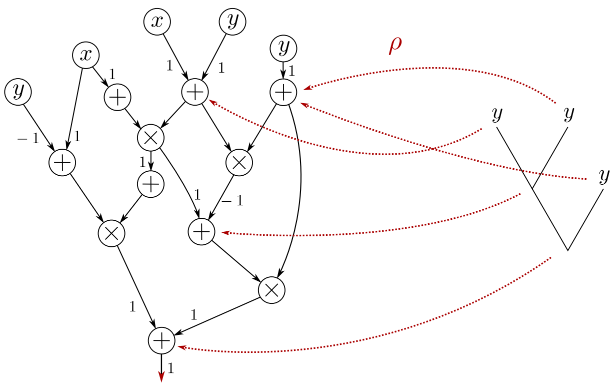


Figure 2.3: A run ρ in the circuit on the left, over the monomial on the right. It has value -1.

LEMMA 2.3. *Let $\mathcal{C}$ be a circuit computing the non-associative polynomial $f : Tree(X) \to K$. Then the coefficient $f(t)$ of a monomial $t \in Tree(X)$ in f is equal to*

$$\sum_{\rho : t \to V_\oplus} val(\rho).$$

We may now state and prove our cornerstone result, which holds in both the commutative and non-commutative settings.

THEOREM 2.4. *Let $f : Tree(X) \to K$ be a non-associative polynomial, H_f be its Hankel matrix, and $\mathcal{C}$ be a circuit computing f. Then $|\mathcal{C}| \geq rank\,(H_f)$. Moreover, if f is homogeneous this bound is tight, meaning there exists a circuit $\mathcal{C}$ computing f of size rank (H_f).*

Note that an interesting feature of this theorem is that the upper bound is effective: given a homogenous polynomial one can construct a circuit computing this polynomial of size rank (H_f).

The proof of the lower bound follows the same lines as Nisan's original proof for non-commutative ABPs (Nisan 1991).

PROOF. We start with the lower bound, that is, $|\mathcal{C}| \geq \text{rank}\,(H_f)$.

Let $\mathcal{C}$ be a circuit computing the non-associative polynomial $f : \text{Tree}(X) \to K$. Let $V_\oplus$ denote the set of addition gates of $\mathcal{C}$.

To bound the rank of the Hankel matrix H_f by $|\mathcal{C}| = |V_\oplus|$ we show that H_f can be written as the sum of $|V_\oplus|$ matrices each of rank at most 1.

For each $v \in V_\oplus$ we define two circuits which decompose the computations around v. Let $\mathcal{C}_1^v$ be the circuit obtained from C by changing all output weights to 0 except that of v which is set to 1. Note that $\mathcal{C}_1^v$ can be seen as the restriction of $\mathcal{C}$ to descendants of v. Let $\mathcal{C}_2^v$ be another copy of $\mathcal{C}$ with just one extra input gate labelled by a fresh variable $\square \notin X$ with a single outgoing edge with weight 1 going to v. We let $f^v : \mathrm{Tree}(X) \to K$ denote the polynomial computed by $\mathcal{C}_1^v$ and $g^v : \mathrm{Context}(X) \to K$ denote the restriction of the polynomial computed by $\mathcal{C}_2^v$ to $\mathrm{Context}(X) \subseteq \mathrm{Tree}(X \sqcup \{\square\})$.

We now show the equality

$$H_f(c,t) = \sum_{v \in V_\oplus} f^v(t) g^v(c).$$

For that, fix a monomial $t \in \mathrm{Tree}(X)$ and a context $c \in \mathrm{Context}(X)$ and denote by $n_\square$ the leaf of c labelled by $\square$, which is also the root of t and the node to which t is substituted with in $c[t]$. Relying on Lemma 2.3, we calculate the coefficient $f(c[t])$ of $c[t]$ in f.

$$\begin{aligned}
f(c[t]) = \sum_{\rho: c[t] \to V_\oplus} \mathrm{val}(\rho) &= \sum_{v \in V_\oplus} \sum_{\substack{\rho: c[t] \to V_\oplus \\ \rho(n_\square) = v}} \mathrm{val}(\rho) \\
&= \sum_{v \in V_\oplus} \sum_{\substack{\rho_1^v: t \to V_\oplus \\ \rho_1^v(n_\square) = v}} \sum_{\substack{\rho_2^v: c \to V_\oplus \\ \rho_2^v(n_\square) = v}} \mathrm{val}(\rho_1^v)\mathrm{val}(\rho_2^v) \\
&= \sum_{v \in V_\oplus} \sum_{\substack{\rho_1^v: t \to V_\oplus \\ \rho_1^v(n_\square) = v}} \mathrm{val}(\rho_1^v) \sum_{\substack{\rho_2^v: c \to V_\oplus \\ \rho_2^v(n_\square) = v}} \mathrm{val}(\rho_2^v) \\
&= \sum_{v \in V_\oplus} f^v(t) g^v(c).
\end{aligned}$$

Let $M_v \in K^{\mathrm{Tree}(X) \times \mathrm{Context}(X)}$ be the matrix given by $M_v(t,c) = f^v(t)g^v(c)$: its rank is at most one as M_v is the product of a column vector by a row vector. The previous equality reads in matrix form

$H_f = \sum_{v \in V_\oplus} M_v$. Hence, we obtain the announced lower bound using rank subadditivity:

$$\operatorname{rank}(H_f) = \operatorname{rank}\left(\sum_{v \in V_\oplus} M_v\right) \leq \sum_{v \in V_\oplus} \operatorname{rank}(M_v) \leq |V_\oplus| = |\mathcal{C}|.$$

We now turn to the upper bound, and assume f is homogeneous.

We first give a construction of a circuit, then provide and prove by induction a strong invariant which implies that the circuit does indeed compute f. For every $t \in \operatorname{Tree}(X)$, we let H_t denote the corresponding column in the Hankel matrix, *i.e.* $H_t : c \mapsto c[t]$.

Let $T \subseteq \operatorname{Tree}(X)$ be such that $(H_t)_{t \in T}$ is a basis of $\{H_t \mid t \in \operatorname{Tree}(X)\}$. In particular T has size $\operatorname{rank}(H_f)$. For any $t' \in \operatorname{Tree}(X)$, we let $\alpha_t^{t'}$ denote the coefficient of H_t in the decomposition of $H_{t'}$ on $(H_t)_{t \in T}$, that is,

$$H_{t'} = \sum_{t \in T} \alpha_t^{t'} H_t. \tag{$\star$}$$

We may now explicitly define circuit $\mathcal{C}$:

- The addition gates are (identified with) elements of T. The output weight of $t \in T$ is $f(t)$.
- The input gates are given by elements of X (and the matching label). The input gate $x \in X$ has an outgoing arc to the addition gate $t \in T$ with weight α_t^x.
- The multiplication gates are given by elements $(t_0, t_1, t) \in T^3$. Such a multiplication gate has an incoming arc from t_0 on the left, an incoming arc from t_1 on the right, and an outgoing arc to t, with weight $\alpha_t^{t_1 \cdot t_2}$.

Note that the size of $\mathcal{C}$ is $|T| = \operatorname{rank}(H_f)$.

For $\mathcal{C}$ to be well-defined as a circuit, it remains to show that its underlying graph is acyclic. This is implied by the fact that $\alpha_t^{t_1 \cdot t_2}$ may only be nonzero if $\deg(t) = \deg(t_1) + \deg(t_2)$, which we now prove. Since f is homogeneous of degree d, H_t may be nonzero only on contexts c such that $\deg(c[t]) = d$, that is, $\deg(c) = d - \deg(t) + 1$. Hence, the set $\{H_t, t \in T\}$ may be partitioned according to the degree of t into parts with disjoint support, so for the decomposition ($\star$) to hold, it must be that $\alpha_t^{t'} \neq 0$ implies $\deg(t) = \deg(t')$.

For $t \in T$, we let $g_t : \mathrm{Tree}(X) \to K$ denote the polynomial computed at gate t in $\mathcal{C}$. We will now show, by induction on the size of $t' \in \mathrm{Tree}(X)$, that $g_t(t') = \alpha_t^{t'}$.

If $t' = x \in X$, then $g_t(t') = \alpha_t^x$, so the base case is clear. We now assume that $t' = t'_1 \cdot t'_2 \in \mathrm{Tree}(X)$, and show that $\sum_{t \in T} g_t(t') H_t = H_{t'}$, which is enough to conclude by uniqueness of the decomposition in($\star$). For that we will show that the previous equality holds for any context $c \in \mathrm{Context}(X)$.

We first remark the following

$$\begin{aligned}
\sum_{t\in T} g_t(t')H_t &= \sum_{t\in T}\left(\sum_{t_1,t_2\in T} \alpha_t^{t_1\cdot t_2} g_{t_1}(t'_1) g_{t_2}(t'_2)\right) H_t \\
&= \sum_{t\in T}\left(\sum_{t_1,t_2\in T} \alpha_t^{t_1\cdot t_2} \alpha_{t_1}^{t'_1}\alpha_{t_2}^{t'_2}\right) H_t \\
&= \sum_{t_1,t_2\in T} \alpha_{t_1}^{t'_1}\alpha_{t_2}^{t'_2}\left(\sum_{t\in T}\alpha_t^{t_1\cdot t_2} H_t\right) \\
&= \sum_{t_1,t_2\in T} \alpha_{t_1}^{t'_1}\alpha_{t_2}^{t'_2} H_{t_1\cdot t_2}.
\end{aligned}$$

Now, let $c \in \mathrm{Context}(X)$. For any tree $t \in \mathrm{Tree}(X)$, we define $c_t^1 = c[\square \cdot t] \in \mathrm{Context}(X)$, and $c_t^2 = c[t \cdot \square] \in \mathrm{Context}(X)$ (see Figure 2.4). Then for any $t_1, t_2, c[t_1 \cdot t_2] = c_{t_2}^1[t_1] = c_{t_1}^2[t_2]$.

Evaluating at c, we now obtain

$$\begin{aligned}
&\sum_{t\in T} g_t(t')H_t(c) \\
&\quad = \sum_{t_1,t_2\in T} \alpha_{t_1}^{t'_1}\alpha_{t_2}^{t'_2} H_{t_1\cdot t_2}(c) = \sum_{t_1,t_2\in T} \alpha_{t_1}^{t'_1}\alpha_{t_2}^{t'_2} f(c[t_1\cdot t_2]) \\
&\quad = \sum_{t_1,t_2\in T} \alpha_{t_1}^{t'_1}\alpha_{t_2}^{t'_2} f(c_{t_2}^1[t_1]) = \sum_{t_1,t_2\in T} \alpha_{t_2}^{t'_2} H_{t_1}(c_{t_2}^1)
\end{aligned}$$

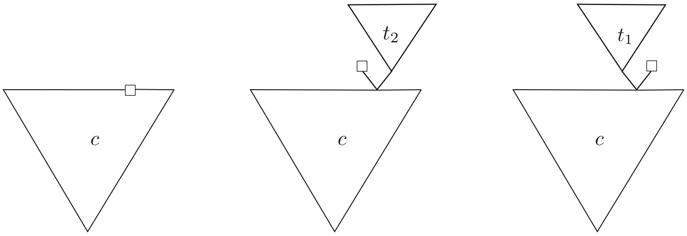


Figure 2.4: A context c, and the contexts $c^1_{t_2}$ and $c^2_{t_1}$.

$$
\begin{aligned}
&= \sum_{t_2 \in T} \alpha_{t_2}^{t'_2} H_{t'_1}(c^1_{t_2}) = \sum_{t_2 \in T} \alpha_{t_2}^{t'_2} H_{t'_1 \cdot t_2}(c) \\
&= \sum_{t_2 \in T} \alpha_{t_2}^{t'_2} f(c^2_{t'_1}[t_2]) = \sum_{t_2 \in T} \alpha_{t_2}^{t'_2} H_{t_2}(c^2_{t'_1}) = H_{t'_2}(c^2_{t'_1}) \\
&= H_t(c)
\end{aligned}
$$

which proves the wanted invariant, namely $g_t(t') = \alpha_t^{t'}$. Hence, the value computed by the circuit for monomial t' is precisely

$$
\sum_{t \in T} g_t(t') f(t) = \sum_{t \in T} \alpha_t^{t'} H_t(\square) = H_{t'}(\square) = f(t'),
$$

which concludes the proof of the upper bound. $\square$

The remainder of this paper consists in applying Theorem 2.4 to obtain lower bounds in various cases. To this end we need a better understanding of the Hankel matrix: in Section 3 we introduce a few concepts and in Section 4 we develop decomposition theorems for the Hankel matrix.

Before digging any deeper we can already give two applications of Theorem 2.4, yielding simple proofs of non-trivial results from the literature.

The first lower bound we obtain is a separation of **VP** and **VNP** in the commutative non-associative setting. It was already obtained in (Hrubeš *et al.* 2010, Theorem 6).

Another early result is an alternative proof of (Arvind & Raja 2016, Theorem 26), which gives an exponential lower bound for the permanent and the determinant against unambiguous circuits in the *associative* setting.

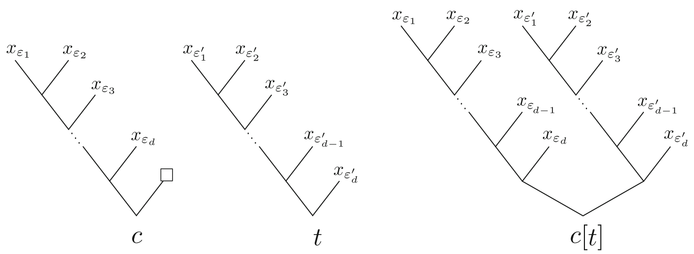


Figure 2.5: The context $c = (((\cdots(((x_{\varepsilon_1} \cdot x_{\varepsilon_2})\ x_{\varepsilon_3})\ x_{\varepsilon_4}) \cdots)\ x_{\varepsilon_d})\square)$, the tree $t = ((\cdots(((x_{\varepsilon'_1} \cdot x_{\varepsilon'_2})\ x_{\varepsilon'_3})\ x_{\varepsilon'_4}) \cdots)\ x_{\varepsilon'_d})$ and their composition $c[t]$.

Separation of commutative non-associative VP and VNP. We now give an alternative separation argument of the classes **VP** and **VNP** in the commutative non-associative setting. The original proof is due to (Hrubeš *et al.* 2010, Theorem 6), it exhibits a polynomial which requires a superpolynomial circuit to be computed. For simplicity, we give a slightly different polynomial, but the proof is very much a reinterpretation of that of Hrubeš *et al.* (2010) in the newly introduced vocabulary.

COROLLARY 2.5. *For $d > 1$, let f be the commutative non-associative polynomial of degree $2d$ and over two variables x_0 and x_1 defined by*

$$f = \sum_{\varepsilon_1,\ldots,\varepsilon_d \in \{0,1\}} (((\cdots(x_{\varepsilon_1} x_{\varepsilon_2}) x_{\varepsilon_3}) \cdots) x_{\varepsilon_d})^2.$$

Any circuit computing f has size at least $3 \times 2^{d-2}$.

PROOF. We give a lower bound on the rank of the Hankel matrix. We consider the submatrix restricted to contexts with $(d+1)$ leaves of the form $(((\cdots(((x_{\varepsilon_1} \cdot x_{\varepsilon_2})\ x_{\varepsilon_3})\ x_{\varepsilon_4}) \cdots)\ x_{\varepsilon_d})\square)$ and to trees with d leaves of the form $((\cdots(((x_{\varepsilon'_1} \cdot x_{\varepsilon'_2})\ x_{\varepsilon'_3})\ x_{\varepsilon'_4}) \cdots)\ x_{\varepsilon'_d})$. See Figure 2.5 for a depiction.

This matrix is a permutation matrix of size $3 \times 2^{d-2}$, which is, up to commutativity, the number of different trees or contexts of the form mentioned above. $\square$

We now present a first lower bound in the *associative* setting. The method we shall use is generic: consider an associative circuit $\mathcal{C}$, from a given restricted class of circuits, computing a given polynomial f. Let $\tilde{f}$ be the non-associative polynomial computed by $\mathcal{C}$ when it is seen as non-associative. The restriction on $\mathcal{C}$ together with the coefficients in f provide information on $\tilde{f}$ which we use to derive a lower bound on rank (H), which is also a lower bound on $\mathcal{C}$ thanks to Theorem 2.4.

Lower bound against associative unambiguous circuits. We give a lower bound for unambiguous circuits computing the *associative* permanent or determinant. A circuit is said ***unambiguous***, if for each (associative) monomial m, there is at most one tree t labelled by m such that $\mathcal{C}$ has a run over t. Such circuits were already studied in Arvind & Raja (2016), in which the authors provide a lower bound for the permanent: we show how to recover their result using the Hankel matrix. Note that this notion makes sense in both the commutative and the non-commutative settings and that our lower bounds hold in both settings.

Recall that, on variables $X = \{x_{i,j} \mid i,j \in [n]\}$, if one lets S_n denote the set of all permutations over $[n]$ and $\mathrm{sgn}(\sigma)$ denote the signature of a permutation σ, the determinant of degree n is the polynomial

$$\mathrm{Det} = \sum_{\sigma \in S_n} \prod_{i=1}^{n} \mathrm{sgn}(\sigma) x_{i,\sigma(i)}$$

and the permanent of degree n is the polynomial

$$\mathrm{Per} = \sum_{\sigma \in S_n} \prod_{i=1}^{n} x_{i,\sigma(i)}.$$

Corollary 2.6. *Any unambiguous circuit computing the determinant or the permanent of degree n has size at least $\binom{n}{n/3}$.*

Proof. Consider an unambiguous circuit $\mathcal{C}$ computing the permanent (the proof is easily adapted to a circuit computing the determinant) of degree n on variables $X = \{x_{i,j} \mid i,j \in [n]\}$. For any permutation σ, let $t_\sigma \in \mathrm{Tree}(X)$ be the unique (non-associative) monomial along which there is a run computing the (associative) monomial $x_{1,\sigma(1)}x_{2,\sigma(2)}\cdots x_{n,\sigma(n)}$. Then, the non-associative polynomial $\tilde{f}$ computed by $\mathcal{C}$ when it is seen as a non-associative circuit is precisely $\tilde{f} = \sum_\sigma t_\sigma$. According to Theorem 2.4, it suffices to lower bound the rank of $H_{\tilde{f}}$.

Let $(A, S) \subseteq [n]^2$ be a pair of subsets. We let $T_{A\to S} \subseteq \mathrm{Tree}(X)$ be the subset of trees t such that the set of first (*resp.* second) indices of the labels of t is precisely A (*resp.* S). Symmetrically, let $C_{A\to S} \subseteq \mathrm{Context}(X)$ be the subset of contexts c such that the set of first (*resp.* second) indices of the labels (except for the $\square$) of c is precisely $[n] \setminus A$ (*resp.* $[n] \setminus S$). If $(A, S) \neq (A', S')$, then $T_{A\to S}$ and $T_{A'\to S'}$ are disjoint, as is the case for $C_{A\to S}$ and $C_{A'\to S'}$. Moreover, if $t \in T_{A\to S}$ and $c \in C_{A'\to S'}$, it must be that $\tilde{f}(c[t]) = 0$. Hence, $H_{\tilde{f}}$ is a block-diagonal matrix, with blocks $H_{A,S}$ being given by restricting the columns to some $T_{A\to S}$ and the rows to $C_{A\to S}$. Note that if $|A| \neq |S|$ then $H_{A,S} = 0$. In particular,

$$\mathrm{rank}\left(H_{\tilde{f}}\right) = \sum_{\substack{A,S\subseteq[n]\\|A|=|S|}} \mathrm{rank}\left(H_{A,S}\right).$$

We now show using a counting argument that an exponential number of such blocks are nonzero and hence, have rank at least 1.

For all permutations σ, we choose a subtree t'_σ of t_σ which has size in $[n/3, 2n/3]$, and let (A_σ, S_σ) be such that $t'_\sigma \in T_{A_\sigma\to S_\sigma}$. Note that $n/3 \leq |A_\sigma| = |S_\sigma| = |t'_\sigma| \leq 2n/3$, and that $H_{A_\sigma,S_\sigma} \neq 0$. Moreover, it must be that $\sigma(A_\sigma) = S_\sigma$. Hence, if $A, S \subseteq [n]$ are fixed such that $n/3 \leq |A| = |S| \leq 2n/3$,

$$|\{\sigma \mid A_\sigma = A \text{ and } S_\sigma = S\}| \leq |\{\sigma \mid \sigma(A) = S\}| \leq \left(\frac{n}{3}\right)! \left(\frac{2n}{3}\right)!$$

Hence, the number of nonzero blocks $H_{A,S}$ is at least

$$\frac{n!}{\left(\frac{n}{3}\right)! \left(\frac{2n}{3}\right)!} = \binom{n}{n/3}$$

which concludes the proof. $\square$

Note that this exact proof goes beyond the case of unambiguous circuits. It is actually sufficient to assume that all non-associative monomials t such that $\tilde{f}(t) \neq 0$ are labelled by a monomial of the form $x_{1,\sigma(1)}x_{2,\sigma(2)} \cdots x_{n,\sigma(n)}$ for some permutation σ.

3. Decomposing the hankel matrix: unique parse tree circuits

Theorem 2.4, as already illustrated by Corollary 2.6, is a natural tool to derive lower bounds thanks to an analysis of the rank of the Hankel Matrix. In order to lower bound this rank for the most general classes possible, we need tools, parse trees and partial derivative matrices, that we introduce now; we then apply these tools to derive a general result regarding the class of Unique Parse Tree circuits (Theorem 3.9). In Section 4, we will push this analysis further and derive generic lower bounds.

3.1. Parse trees. With any monomial $t \in \mathrm{Tree}(X)$ we associate its ***shape*** $\mathrm{shape}(t) \in \mathrm{Tree}$ as the tree obtained from t by removing the labels at the leaves.

DEFINITION 3.1. *Let $\mathcal{C}$ be a circuit computing a non-commutative non-associative polynomial f. A **parse tree** of $\mathcal{C}$ is any shape $s \in$ Tree for which there exists a monomial $t \in$ Tree(X) whose coefficient in f is nonzero and such that $s =$ shape(t). We let $PT(\mathcal{C}) = \{$shape$(t) \mid f(t)$ nonzero$\}$.*

The notion of parse trees has been considered in many previous works, see for example Allender *et al.* (1998); Arvind & Raja (2016); Jerrum & Snir (1982); Lagarde *et al.* (2018, 2016); Malod & Portier (2008); Saptharishi & Tengse (2017).

REMARK 3.2. *Let $\mathcal{C}$ be a circuit computing a homogeneous polynomial of degree d. Then asymptotically, $|PT(\mathcal{C})| \leq 4^d$. Indeed, the maximal number of parse trees corresponds to the number of ordered binary trees with d leaves which is the $(d-1)$-th Catalan number C_{d-1}. Asymptotically, one has $C_k \sim \frac{4^k}{k^{3/2}\sqrt{\pi}}$ which implies the announced lower bound on the number of parse trees.*

3.2. Partial derivative matrices. We now introduce a popular tool for proving circuit lower bounds, namely, partial derivative matrices, originated from Hyafil (1977); Nisan (1991) and widely used and extended in subsequent works, see for example Dvir *et al.* (2012); Gupta *et al.* (2014); Kayal *et al.* (2014a); Kumar & Saraf (2017); Limaye *et al.* (2016); Nisan & Wigderson (1997).

For $A \subseteq [d]$ of size i ,$u \in X^{d-i}$, and $v \in X^i$, we define the monomial $u \otimes_A v \in X^d$: it is obtained by interleaving u and v with u taking the positions indexed by $[d] \setminus A$ and v the positions indexed by A. For instance $x_1x_2 \otimes_{\{2,4\}} y_1y_2 = x_1y_1x_2y_2$.

Definition 3.3. *Let f be a homogeneous non-commutative associative polynomial. Let $A \subseteq [d]$ be a set of positions of size i.*

*The **partial derivative matrix** $M_A(f)$ of f with respect to A is defined as follows: the rows are indexed by $u \in X^{d-i}$ and the columns by $v \in X^i$, and the value of $M_A(f)(u,v)$ is the coefficient of the monomial $u \otimes_A v$ in f.*

Remark 3.4. *The terminology* partial derivative matrix, *widely adopted in the literature, comes from the observation that the row labelled by monomial u of the matrix contains the coefficients of the partial derivative $\frac{\partial f}{\partial u}$. The same remark can be made for columns of $M_A(f)$. This will not be exploited in this paper.*

Example 3.5. Let $f = xyxy + 3xxyy + 2xxxy + 5yyyy$ and $A = \{2,4\}$. Then $M_A(f)$ is given below.

	$x\,x$	$x\,y$	$y\,x$	$y\,y$
$x\,x$	0	2	0	1
$y\,x$	0	0	0	0
$x\,y$	0	3	0	0
$y\,y$	0	0	0	5

$\Diamond$

We define a distance dist : $\mathcal{P}([d]) \times \mathcal{P}([d]) \to \mathbb{N}$ on subsets of $[d]$ by letting $\mathrm{dist}(A, B)$ be the minimal number of additions and deletions of elements of $[d]$ to go from A to B, assuming that complementing is for free. Formally,

$$\operatorname{dist}(A, B) = \min\{|\Delta(A, B)|, |\Delta(A^c, B)|\},$$

where $\Delta(A, B) = (A \setminus B) \cup (B \setminus A)$ is the symmetric difference between A and B. This is illustrated in Figure 3.1.

Remark 3.6. *A similar looking notion of distance is also available in the current literature for commutative depth-4 lower bounds. This was first implicitly defined by Fournier et al. (2014) and by Kayal et al. (2014a), and later made explicit by Chillara & Mukhopadhyay (2019).*

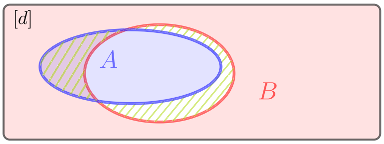


Figure 3.1: In this case, the symmetric difference is smaller when complementing one of the sets, so $\operatorname{dist}(A, B)$ is the cardinality of the hatched subset.

Remark 3.7. *The apparent asymmetry in the definition is artificial as it does hold that $\operatorname{dist}(A, B) = \operatorname{dist}(B, A)$. It is also the case that $\operatorname{dist}(A, B) = 0 \implies A = B$ or $A = B^c$. In fact dist is indeed a distance over subsets of $[d]$ modulo complementation.*

The following lemma (see e.g., Limaye *et al.* 2016) informally says that, if A and B are close to each other, then the ranks of the corresponding partial derivative matrices are close to each other as well. Though it is well known, we give a proof for completeness.

Lemma 3.8. *Let f be a homogeneous non-commutative associative polynomial of degree d with n variables. Then, for any subsets $A, B \subseteq [d]$, $\operatorname{rank}(M_A(f)) \leq n^{\operatorname{dist}(A,B)} \operatorname{rank}(M_B(f))$.*

Proof. Without loss of generality, one may safely assume that $\mathrm{dist}(A,B) = |\Delta(A,B)|$ (by transposing the matrix $M_A(f)$ if necessary).

We prove the statement by induction on $d = |\Delta(A,B)|$. If $d = 0$, this is trivial since A and B are identical in this case. For the case $d = 1$, let us assume that $A = B \cup \{i\}$ (the other case being very similar). We divide $M_A(f)$ into horizontal blocks, one for each variable x, that we call $M_A(f)^x$, corresponding to the monomials for which the position i is occupied by the variable x. Therefore the rank of $M_A(f)$ is upper bounded by $\sum_x \mathrm{rank}(M_A(f)^x)$, but each $M_A(f)^x$ is a submatrix of $M_B(f)$ so that $\mathrm{rank}(M_A(f)^x) \leq \mathrm{rank}(M_B(f))$, hence the result.

If $d > 1$, we first find a set C such that $|\Delta(A,C)| = 1$ and $|\Delta(C,B)| = d-1$, and we conclude by applying the induction hypothesis and using the case $d = 1$. $\square$

At this point, we have the material in hands to describe a precise characterization of the size of the smallest Unique Parse Tree circuit which computes a given polynomial. We take this short detour before moving on to our core lower bound results in Section 4.

3.3. Characterization of smallest unique parse tree circuit. Unique Parse Tree (UPT) circuits are non-commutative associative circuits with a unique parse tree. They were first introduced in Lagarde *et al.* (2016). They generalize ABPs, which are equivalent to UPT circuits with a left comb as their unique parse tree (a left comb being a tree corresponding to the shape of tree t in Figure 2.5). Hence, we recover Nissan's Theorem (Nisan 1991) when instantiating our characterization result, Theorem 3.9, to left combs. Our techniques allow a slight improvement and a better understanding of their results since the original result requires a normal form which can lead to an exponential blow-up.

Given a shape $s \in \mathrm{Tree}$ of size d, i.e., with d leaves and a node v of s, we let s_v denote the subtree of s rooted in v, and $I_v \subseteq [d]$ denote the interval of positions of the leaves of s_v in s. We say that $s' \in \mathrm{Tree}$ is a ***subshape*** of s if $s' = s_v$ for some v, and that $I \subseteq [d]$ is spanned by s if $I = I_v$ for some v. Figure 3.2 illustrates the occurrences of a subshape in a shape.

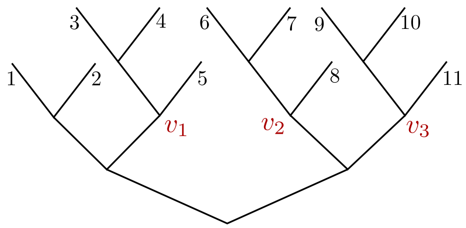


Figure 3.2: A shape of size 11, in which three nodes v_1, v_2, v_3 span the same subshape. The corresponding spanned intervals are $I_{v_1} = [3,5], I_{v_2} = [6,8]$ and $I_{v_3} = [9,11]$. We display the position of each leaf for readability.

Let f be a homogeneous non-commutative associative polynomial of degree d, let $s \in$ Tree be a shape of size d, and let s' be a subshape of s such that $v_1, \ldots, v_p$ are all the nodes v of s such that $s' = s_v$. We define

$$M_{s'} = \begin{bmatrix} M_{I_{v_1}}(f) \\ M_{I_{v_2}}(f) \\ \vdots \\ M_{I_{v_p}}(f) \end{bmatrix}.$$

THEOREM 3.9. *Let f be a homogeneous non-commutative associative polynomial of degree d and let $s \in$ Tree be a shape of size d. Then the smallest UPT circuit with shape s computing f has size exactly*

$$\sum_{s' \text{ subshape of } s} \mathit{rank}\,(M_{s'})\,.$$

PROOF. Let $\mathcal{C}$ be a UPT circuit with shape s computing f. We let $\tilde{f}$ denote the non-associative polynomial computed by $\mathcal{C}$.

Since $\mathcal{C}$ is UPT with shape s, $\tilde{f}$ is the *unique* non-associative polynomial which is nonzero only on trees with shape s and projects to f, i.e., $\tilde{f}(t) = f(u)$ if shape$(t) = s$ and t is labelled by u, and $\tilde{f}(t) = 0$ otherwise.

In particular, the size of the smallest UPT circuit with shape s computing f is the same as the size of the smallest circuit computing $\tilde{f}$, which thanks to Theorem 2.4 is equal to the rank of the Hankel matrix $H_{\tilde{f}}$.

The Hankel matrix of $\tilde{f}$ may be nonzero only on columns indexed by trees whose shapes s' are subshapes of s, and on such columns, nonzero values are on rows corresponding to a context obtained from s by replacing an occurrence of s' by $\square$. The corresponding blocks are precisely the matrices $M_{s'}$, and are placed in a diagonal fashion, hence the lower bound. $\square$

Theorem 3.9 can be applied to concrete polynomials, for instance to the permanent of degree d.

COROLLARY 3.10. *Let $s \in$ Tree be a shape. The smallest UPT circuit with shape s computing the permanent has size*

$$\sum_{v \text{ node of } s} \binom{d}{|I_v|},$$

where I_v is the set of leaves in the subtree rooted at v in s. In particular, this is always larger than $\binom{d}{d/3}$.

PROOF. Let s' be a subshape of s, and $v_1, ..., v_p$ be all the nodes of s such that $s_{v_i} = s'$. Let $\ell = |I_{v_i}|$ which does not depend on i. There are no $i \neq j$ such that v_i is a descendant of v_j, so the I_{v_i} are pairwise disjoint. Let $I_{v_i} = [a_i, a_i + \ell - 1]$. The coefficient of $M_{I_{v_i}}(\mathrm{Per})$ in $(u, w) \in X^{d-\ell} \times X^{\ell}$, namely, $\mathrm{Per}(u \otimes_{I_{v_i}} w)$, may be nonzero only if w is of the form

$$x_{a_i,b_1} x_{a_i+1,b_2} \cdots x_{a_i+\ell-1,b_\ell}$$

for some $b_1, \ldots, b_\ell \in [d]$. In particular, the $M_{I_{v_i}}(\mathrm{Per})$ have nonzero columns with disjoint supports, so

$$\mathrm{rank}\,(M_{s'}) = \sum_i \mathrm{rank}\left(M_{I_{v_i}}(\mathrm{Per})\right).$$

We claim now that $\mathrm{rank}\left(M_{I_{v_i}}(\mathrm{Per})\right) = \binom{d}{\ell}$, which leads to the announced formula. Indeed, any subset A of $[d]$ of size ℓ corresponds to a block full of 1's in the matrix $M_{I_{v_i}}(\mathrm{Per})$ in the following way: $\mathrm{Per}(u \otimes_{I_{v_i}} w) = 1$ whenever u is a monomial whose first indices are $[d] \setminus I_{v_i}$ and the second indices are any permutation of $[d] \setminus A$, and w is a monomial whose first indices are I_{v_i} and the second indices are any permutation of A. Two such blocks have disjoint rows and columns, and these are the only 1's in $M_{I_{v_i}}(\mathrm{Per})$. Moreover, there are $\binom{d}{\ell}$ such sets A.

$\square$

Applied to s being a left-comb, Corollary 3.10 yields that the smallest ABP computing the permanent has size $2^d + d$. Applied to s being a complete binary tree of depth $k = \log d$, the size of the smallest UPT is $\Theta\left(\frac{2^d}{d}\right)$.

4. Decomposing the Hankel matrix: generic lower bounds

We now get to the technical core of the paper where we establish generic lower bound theorems through a decomposition of the Hankel matrix, that we will later instantiate in Section 5 to concrete classes of circuits.

We first restrict ourselves to the non-commutative setting. Our first decomposition, Theorem 4.1, seems to capture mostly previously known techniques. However, the second, more powerful, decomposition, Theorem 4.2, takes advantage of the global shape of the Hankel matrix. Doing so allows to go beyond previous results only hinging around considering partial derivatives matrices which turn out to be isolate slices of the Hankel matrix.

We later explain in Section 4.3 how to extend the study to the commutative case.

4.1. General roadmap. Let f be a (commutative or non-commutative) associative polynomial for which we want to prove lower bounds. Consider a circuit $\mathcal{C}$ which computes f, and let $\tilde{f}$ be the non-associative polynomial computed by $\mathcal{C}$. Our aim is, following Theorem 2.4, to lower bound the rank of the Hankel matrix $H_{\tilde{f}}$. We know that polynomials $\tilde{f}$ and f are equal up to associativity, which provides linear relations among the coefficients of $H_{\tilde{f}}$.

The bulk of the technical work is to reorganize the rows and columns of $H_{\tilde{f}}$ in order to decompose it into blocks which may be identified as partial derivative matrices with respect to some subsets $A_1, A_2, \cdots \subseteq [d]$, of some associative polynomials which depend on $\tilde{f}$ and sum to f. The number and choice of these subsets depend on the parse trees of the circuit $\mathcal{C}$.

Now, assume that there exists a subset $A \subseteq [d]$ which is at distance at most δ to each A_i. Losing a factor of n^δ on the rank through the use of Lemma 3.8 we reduce the aforementioned blocks of $H_{\tilde{f}}$ to partial derivatives with respect to A. Such matrices can then be summed to recover the partial derivative matrix of f with respect to A, yielding in the lower bound a (dominating) factor of $\operatorname{rank}(M_A(f))$.

4.2. Generic lower bounds in the non-commutative setting. Following the general roadmap described above, we obtain a first generic lower bound result.

THEOREM 4.1. *Let f be a non-commutative homogeneous polynomial of degree d computed by a circuit $\mathcal{C}$. Let $A \subseteq [d]$ and $\delta \in \mathbb{N}$ such that all parse trees of $\mathcal{C}$ span an interval at distance at most δ from A. Then $\mathcal{C}$ has size at least $\operatorname{rank}(M_A(f))\, n^{-\delta} |PT(\mathcal{C})|^{-1}$.*

PROOF. The proof relies on a better understanding of the structure of the Hankel matrix $H = H_{\tilde{f}}$ of a general non-associative polynomial $\tilde{f} : \mathrm{Tree}(X) \to K$.

More precisely, we organize the columns and rows of H in order to write it as a block matrix in which we can identify and understand the blocks in terms of partial derivative matrices of some non-commutative (but associative) polynomials which will eventually correspond to parse trees. In the following we refer to Figure 4.1 for illustration of the decompositions.

Recall that $\mathrm{Tree}_k(X) \subseteq \mathrm{Tree}(X)$ denotes the set of trees with k leaves, and let $\mathrm{Context}_k(X) \subseteq \mathrm{Context}(X)$ denote the set of contexts with k leaves (among which one is labelled by $\square$). Note that any tree $t \in \mathrm{Tree}_d(X)$ decomposes into $2d-1$ different pairs $(t', c) \in \mathrm{Tree}_k(X) \times \mathrm{Context}_{d-k+1}(X)$ for some k, such that $c[t'] = t$, which correspond to the $2d-1$ nodes in t. We further partition $\mathrm{Context}_k(X) = \bigcup_{p=1}^{k} \mathrm{Context}_k^p(X)$, with $\mathrm{Context}_k^p(X)$ being the set of contexts where $\square$ is on the p-th leaf.

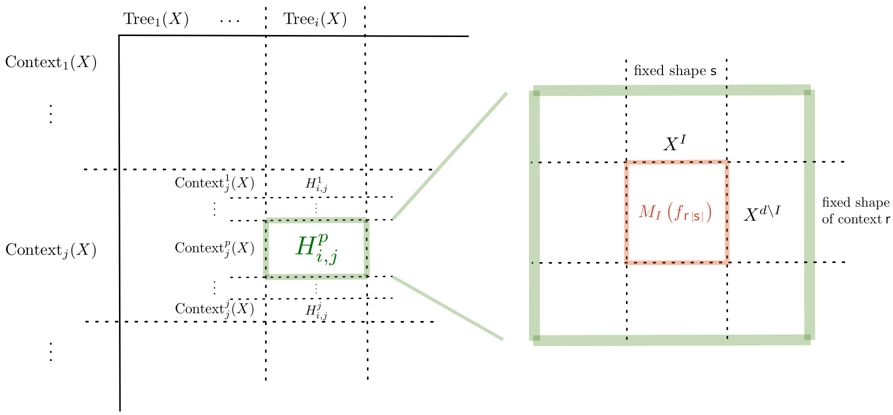


Figure 4.1: Decomposing H as blocks $H_{i,j}^p$, which further decompose into partial derivative matrices. Here, I denotes the interval $[p, p+i-1]$.

Using these partitions for trees and contexts, we may write H as a block matrix with blocks $H_{i,j} = H_{|\mathrm{Tree}_i(X)\times\mathrm{Context}_j(X)}$. Using the finer refinement of contexts, we write block $H_{i,j}$ as a tower (recall that contexts label the rows of H) of sub-blocks $H_{i,j}^p$, for $p \in [j]$, where $H_{i,j}^p = H_{|\mathrm{Tree}_i(X)\times\mathrm{Context}_j^p(X)}$. We now focus on $H_{i,j}^p$, which we will further decompose into blocks that are partial derivative matrices of some homogeneous non-commutative polynomials on the interval $[p, p+i-1]$.

As $\mathrm{Tree}_i(X)$ is the set of trees with i leaves, it can be seen as all possible labeling of shapes with i leaves by variables in X. Hence, $\mathrm{Tree}_i(X) \simeq \mathrm{Tree}_i \times X^i \simeq \mathrm{Tree}_i \times X^{[p,p+i-1]}$. Likewise, $\mathrm{Context}_j^p(X)$ is the set of contexts with j leaves and $\Box$ on the p-th leave, which can be seen as $\mathrm{Context}_j^p(X) \simeq \mathrm{Context}_j^p \times X^{j-1} \simeq \mathrm{Context}_j^p \times X^{[1,i+j-1]\setminus[p,p+i-1]}$, where $\mathrm{Context}_j^p$ is the set of contexts of size j with no labels, except for a unique $\Box$ on the p-th leaf. We now let, for any shape $s \in \mathrm{Tree}_{i+j-1}$, the non-commutative (but associative) homogeneous polynomial f_s of degree $i+j-1$ be defined by

$$\begin{array}{rcl} f_s : X^{i+j-1} & \to & K \\ u & \mapsto & \tilde{f}(s \text{ labelled by } u) \end{array}$$

Now, grouping the columns $t \in \mathrm{Tree}_i(X)$ of $H^p_{i,j}$ which correspond to the same shape $s \in \mathrm{Tree}_i$, and the rows $c \in \mathrm{Context}^p_j(X)$ which correspond to the same shape (of context) $r \in \mathrm{Context}^p_j$, we obtain a block matrix, in which the block indexed by (s,r) is precisely the partial derivative matrix $M_{[p,p+i-1]}\left(f_{r[s]}\right)$.

In the following, we will be interested in non-associative polynomials $\tilde{f} : \mathrm{Tree}(X) \to K$ which project to a given associative $f : X^* \to K$, meaning that for each $u \in X^*$,

$$\sum_{\substack{t \in \mathrm{Tree}(X) \\ \mathrm{label}(t) = u}} \tilde{f}(t) = f(u).$$

In this setting, one can see the decomposition $f = \sum_{s \in \mathrm{Tree}} f_s$ as a decomposition over parse trees of a circuit computing f, f_s being the contribution of the parse tree s in the computation of f. We have seen that if $I = [p, p+i-1]$ is an interval such that s decomposes into $s = r[s']$ for $(s', r) \in \mathrm{Tree}_i \times \mathrm{Context}^p_j$, which means that I is spanned by s, then $M_I(f_s)$ appears as a sub-matrix of H. Hence,

$$(\star) \qquad \max_{\substack{s \in \mathrm{Tree} \\ I \text{ spanned by } s}} \mathrm{rank}\left(M_I\left(f_s\right)\right) \quad \leq \quad \mathrm{rank}\left(H\right).$$

Now, we have all the necessary tools to prove Theorem 4.1. Let $\tilde{f} : \mathrm{Tree}(X) \to K$ be the non-associative polynomial computed by $\mathcal{C}$ when it is seen as a non-associative circuit. For any shape $s \in \mathrm{Tree}_d$, let $f_s : X^d \to K$ be defined as previously. In particular, $\sum_{s \in \mathrm{PT}(\mathcal{C})} f_s = f$.

With a shape $s \in \mathrm{PT}\left(\mathcal{C}\right)$, we associate an interval $I(s)$ spanned by s and such that $\mathrm{dist}(A, I(s)) \leq \delta$. Then we have

$$\begin{aligned}
\mathrm{rank}\left(M_A\left(f\right)\right) &= \mathrm{rank}\left(\sum_{s \in \mathrm{PT}(\mathcal{C})} M_A\left(f_s\right)\right) \\
&\leq \sum_{s \in \mathrm{PT}(\mathcal{C})} \mathrm{rank}\left(M_A\left(f_s\right)\right) \quad \text{by rank subadditivity} \\
&\leq \sum_{s \in \mathrm{PT}(\mathcal{C})} n^\delta \mathrm{rank}\left(M_{I(s)}\left(f_s\right)\right) \quad \text{by Lemma 3.8} \\
&\leq \left|\mathrm{PT}\left(\mathcal{C}\right)\right| n^\delta \mathrm{rank}\left(H\right) \quad \text{by equation } (\star)
\end{aligned}$$

Since, by Theorem 2.4, $\operatorname{rank}(H) \geq \operatorname{rank}(M_A(f))\, n^{-\delta} |\mathrm{PT}(\mathcal{C})|^{-1}$ is a lower bound on $|\mathcal{C}|$, we obtain the announced result. $\square$

The crux to prove Theorem 4.1 is to identify for each parse tree s of $\mathcal{C}$ a block in $H_{\tilde{f}}$ containing the partial derivative matrix $M_{I(s)}(f_s)$ where f_s is the polynomial corresponding to the contribution of the parse tree s in the computation of f and $I(s)$ is an interval spanned by s.

However, we do not consider in this analysis how these blocks are located relative to each other. A more careful analysis of $H_{\tilde{f}}$ consists in grouping together all parse trees that lead to the same spanned interval. Aligning and then summing these blocks we replace the dependence in $|\mathrm{PT}(\mathcal{C})|$ by d^2 which corresponds to the total number of possibly spanned intervals of $[d]$. This yields Theorem 4.2.

Theorem 4.2. *Let f be a non-commutative homogeneous polynomial of degree d computed by a circuit $\mathcal{C}$. Let $A \subseteq [d]$ and $\delta \in \mathbb{N}$ such that all parse trees of $\mathcal{C}$ span an interval at distance at most δ from A. Then $\mathcal{C}$ has size at least $\operatorname{rank}(M_A(f))\, n^{-\delta} d^{-2}$.*

Remark 4.3. *Note that this is an important improvement since the number of parse trees can be up to about 4^d (as noticed in Remark 3.2). As we shall see in Section 5 the lower bounds we obtain using Theorem 4.1 match known results, while using Theorem 4.2 yields substantial improvements.*

Before going on to the formal proof of Theorem 4.2, we start by giving a high-level interpretation of the techniques used to go from Theorem 4.1 to Theorem 4.2. Our aim is still to lower bound the rank of the Hankel matrix $H = H_{\tilde{f}}$ of some (unknown) non-associative polynomial $\tilde{f}$, under the constraints that, for each $u \in X^*$,

$$\sum_{\substack{t \in \mathrm{Tree}(X) \\ \mathrm{label}(t) = u}} \tilde{f}(t) = f(u),$$

for some non-commutative (but associative) polynomial $f : X^* \to K$ that we control. Given the form of our constraints, a natural strategy would be to sum some well chosen sub-matrices of H in order to obtain a matrix that depends only on f, which we could choose to have high rank.

As exposed earlier when proving Theorem 4.1, it is possible to decompose f as the sum of some f_s's, where s ranges over the shapes used by $\tilde{f}$, and then obtain partial derivative matrices of the f_s's with respect to interval spanned by s, as sub-matrices of H. If one can find a subset $A \subseteq [d]$ such that each s spans an interval $I(s)$ that is δ-close to A for some small δ, then one obtains a lower bound for polynomials f with high rank with respect to A.

This first method leads to Theorem 4.1 and it is already strong enough to prove several lower bounds. We believe that in many occurrences in the literature, when obtaining lower bounds involving a circuit decomposition and a partial derivative matrix with respect to a given partition of the set of positions $[d]$, this is somehow the underlying method.

However, this method poorly makes use of the structure of H, since it may happen that some of the chosen sub-blocks are face to face with one another. A short illustration of this phenomenon is the following. Let

$$M = \left(\begin{array}{cc|cc} A_{1,1} & A_{1,2} & \multicolumn{2}{c}{\multirow{2}{*}{C_1}} \\ A_{2,1} & A_{2,2} & & \\ \hline \multicolumn{2}{c|}{\multirow{2}{*}{C_2}} & B_{1,1} & B_{1,2} \\ & & B_{2,1} & B_{2,2} \end{array}\right)$$

be a block matrix, for which one wants to obtain a lower bound on the rank, knowing a lower bound on rank $\left(\sum_{i,j} A_{i,j} + B_{i,j}\right)$, and with no assumption on the C_i's.

The previous method would go as follows:

$$\begin{aligned} \operatorname{rank}(M) &\geq \max\left[\max_{i,j} \operatorname{rank}(A_{i,j}), \max_{i,j} \operatorname{rank}(B_{i,j})\right] \\ &\geq \frac{1}{8} \sum_{i,j} \operatorname{rank}(A_{i,j}) + \operatorname{rank}(B_{i,j}) \\ &\geq \frac{1}{8} \operatorname{rank}\left(\sum_{i,j} A_{i,j} + B_{i,j}\right). \end{aligned}$$

Note that we have lost a factor of 8, which is the number of small blocks that we wish to sum.

A more efficient method would consist in first summing rows and columns of M in order to put together the A's and the B's. This would go as follows, for some matrices C'_1 and C'_2,

$$\begin{aligned}\operatorname{rank}(M) &\geq \operatorname{rank}\left(\begin{bmatrix}\sum_{i,j} A_{i,j} & C'_1 \\ C'_2 & \sum_{i,j} B_{i,j}\end{bmatrix}\right) \\ &\geq \max\left[\operatorname{rank}\left(\sum_{i,j} A_{i,j}\right), \operatorname{rank}\left(\sum_{i,j} B_{i,j}\right)\right] \\ &\geq \frac{1}{2}\operatorname{rank}\left(\sum_{i,j} A_{i,j} + B_{i,j}\right).\end{aligned}$$

By doing so, we have decreased the factor 8 to 2, which is the number of larger blocks.

Back to the Hankel matrix H, this corresponds to putting together the polynomials f_s for which we have chosen the same spanned interval (this corresponds to d^2 larger blocks) instead of considering them separately (which corresponds to $|\mathrm{PT}(\mathcal{C})|$ smaller blocks). We now formalize this idea, using a total order to model the choice of intervals for convenience.

Lemma 4.4. *Let $\tilde{f}$: Tree$(X) \to K$ be a non-associative non-commutative polynomial and let $\leq_{int}$ be a total order on intervals of $[d]$. For any shape s, let $I(s)$ be the smallest (with respect to $\leq_{int}$) interval spanned by s. For any interval I, define a non-commutative associative polynomial by*

$$\begin{aligned} f_I : X^* &\to K \\ u &\mapsto \sum_{\substack{t\in Tree(X)\\ label(t)=u \\ I(shape(t))=I}} \tilde{f}(t).\end{aligned}$$

Then, one has $\max_I \operatorname{rank}(M_I(f_I)) \leq \operatorname{rank}(H_{\tilde{f}})$.

We illustrate the definition of f_I through a small example. Let $t = ((xy)z)$, and assume $[1,2]$ is the smallest interval spanned by

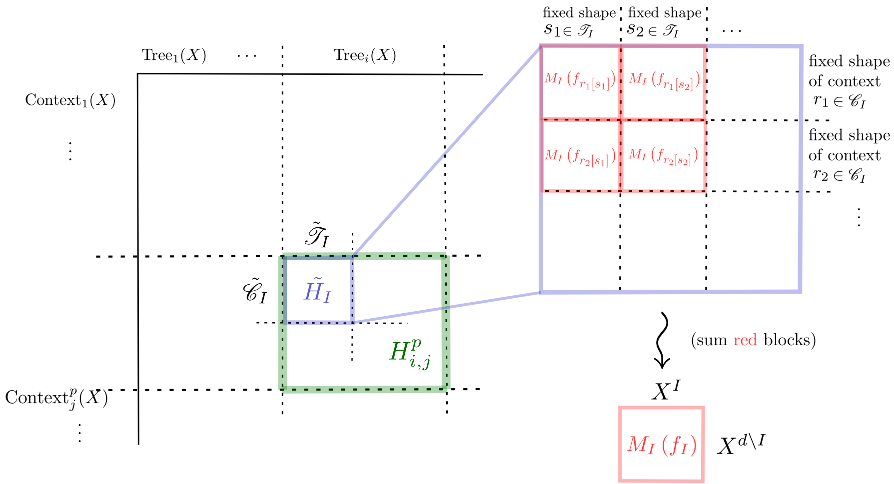


Figure 4.2: Decomposition of the Hankel matrix used in the proof of Lemma 4.4. Here, $I = [p, p + i - 1]$.

t, that is, $[1, 2] \leq_{int} \{1\}, \{2\}, \{3\}, [1, 3]$. Then $\tilde{f}(t)$ will contribute to $f_{[1,2]}(xyz)$ as label$(t) = xyz$ and $I(\text{shape}(t)) = [1, 2]$.

PROOF. Our aim is to obtain $M_I(f_I)$ from $H_{\tilde{f}}$, by first taking a sub-matrix, then adequately summing its rows and columns. The proof is summarized in Figure 4.2.

Let $I = [p, p + i - 1]$ be some fixed interval and $j = d - i + 1$.

Let $r \in \text{Context}^p_j$ be a shape of a context of size j and where $\square$ is on the p-th leaf, let v be a node in r and let $[a, b]$ be the interval spanned by v in r. We define the interval I'_v by

$$I'_v = \begin{cases} [a, b] \text{ if } b < p \\ [a, b + i - 1] \text{ if } a \leq p \leq b \\ [a + i - 1, b + i - 1] \text{ if } a > p, \end{cases}$$

The interval I'_v is to be seen as the interval of positions of the leaves that are descendants of v in some $r[s']$ where s' is any element of Tree$_i$. In particular, if v is the leaf labelled by $\square$ in r, then $I'_v = I$.

Likewise, for a node v of a (sub)shape $s' \in \text{Tree}_i$, we define I'_v by $I'_v = [a + p - 1, b + p - 1]$, where $[a, b]$ is the interval spanned by v in s'. Note that if v is the root of s' then $I_v = I$.

We may now define (we use order $\leq_{int}$ on intervals)

$$\mathscr{C}_I = \{r \in \mathrm{Context}_j^p \mid I = \min_{v \text{ node in } r} I'_v\},$$

and

$$\mathscr{T}_I = \{s' \in \mathrm{Tree}_i \mid I = \min_{v \text{ node in } s'} I'_v\}.$$

We extend these subsets to labelled trees and context in a straightforward fashion by defining $\tilde{\mathscr{C}}_I = \{c \in \mathrm{Context}_j^p(X) \mid \mathrm{shape}(c) \in \mathscr{C}_I\}$ and $\tilde{\mathscr{T}}_I = \{t \in \mathrm{Tree}_i(X) \mid \mathrm{shape}(t) \in \mathscr{T}_I\}$.

Remark that for any $t \in \mathrm{Tree}(X)$ and $u \in X^*$, one has $\mathrm{label}(t) = u$ and $I = I(\mathrm{shape}(t))$ if and only if $t = r[s]$ for some $(s, r) \in \tilde{\mathscr{T}}_I \times \tilde{\mathscr{C}}_I$ such that $u = \mathrm{label}(s) \otimes_I \mathrm{label}(c)$.

We now consider the submatrix $\tilde{H}_I$ of $H_{i,j}^p$ where the rows are restricted to $\tilde{\mathscr{C}}_I$ and the columns to $\tilde{\mathscr{T}}_I$. In this matrix, we now sum the rows which are indexed by contexts with the same label, and the columns which are indexed by trees with the same label, to obtain matrix H_I. Clearly, $\mathrm{rank}\,(H_I) \leq \mathrm{rank}\left(H_{\tilde{f}}\right)$. We finally prove that $H_I = M_I\,(f_I)$. Indeed, let $g \in X^I \simeq X^i$ and $h \in X^{d \setminus A} \simeq X^j$. Then

$$M_I\,(f_I)\,(g,h) = \sum_{\substack{t \in \mathrm{Tree}(X) \\ \mathrm{label}(t) = g \otimes_I h \\ I(\mathrm{shape}(t)) = I}} \tilde{f}(t) = \sum_{\substack{s \in \tilde{\mathscr{T}}_I \\ c \in \tilde{\mathscr{C}}_I \\ \mathrm{label}(s) = g \\ \mathrm{label}(c) = h}} \tilde{f}(c[s]) = H_I(g,h),$$

which concludes the proof of Lemma 4.4. $\square$

With Lemma 4.4 in hands, we are ready to prove Theorem 4.2. Let $\tilde{f} : \mathrm{Tree}(X) \to K$ be the non-associative polynomial computed by $\mathcal{C}$ when seen as a non-associative circuit. Let $\leq_{int}$ be a total order on intervals of d such that $I \mapsto \mathrm{dist}(I, A)$ is non-decreasing. In other words, $I_1 <_{int} I_2$ if and only if $d(I_1, A) < d(I_2, A)$. Let $f_I : X^d \to K$ be given by

$$f_I(u) = \sum_{\substack{t \in \mathrm{Tree}(X) \\ \mathrm{label}(t) = u \\ I(\mathrm{shape}(t)) = I}} \tilde{f}(t).$$

Then any interval I such that $d(I, A) > \delta$ is such that for every parse tree $s \in \mathrm{PT}\,(\mathcal{C})$, one has $I \neq I(s)$, so $f_I = 0$. Hence, we obtain

$$\begin{aligned}
&\operatorname{rank}\left(M_A\left(f\right)\right)\\
&\quad=\operatorname{rank}\left(M_A\left(\sum_{I\text{ interval of }[d]} f_I\right)\right)\\
&\quad=\operatorname{rank}\left(M_A\left(\sum_{\substack{I\text{ interval of }[d]\\ \operatorname{dist}(A,I)\le\delta}} f_I\right)\right)\\
&\quad\le\sum_{\substack{I\text{ interval of }[d]\\ \operatorname{dist}(A,I)\le\delta}}\operatorname{rank}\left(M_A\left(f_I\right)\right)\qquad\text{by rank subadditivity}\\
&\quad\le\sum_{\substack{I\text{ interval of }[d]\\ \operatorname{dist}(A,I)\le\delta}} n^\delta\operatorname{rank}\left(M_I\left(f_I\right)\right)\qquad\text{by Lemma 3.8}\\
&\quad\le d^2n^\delta\operatorname{rank}\left(H_{\tilde f}\right)\qquad\text{by Lemma 4.4}
\end{aligned}$$

which yields the announced lower bound.

4.3. General lower bounds in the commutative setting. We explain how to extend the notions of parse trees and the generic lower bound theorems to the associative commutative setting.

Let $X = X_1 \sqcup X_2 \sqcup \cdots \sqcup X_d$ be a partition of the set X of variables. A monomial is ***set-multilinear*** with respect to the partition if it is the product of exactly one variable from each set X_i, and a polynomial is set-multilinear if all its monomials are.

Example 4.5. The permanent and the determinant of degree d are set-multilinear with respect to the partition $X = X_1 \sqcup X_2 \sqcup \cdots \sqcup X_d$ where $X_i = \{x_{i,j}, j \in [d]\}$. The iterated matrix multiplication polynomial is another example of an important and well-studied set-multilinear polynomial. $\diamondsuit$

Partial derivative matrices also make sense in the realm of set-multilinear polynomials.

Definition 4.6. *Let $X = X_1 \sqcup X_2 \sqcup \cdots \sqcup X_d$, f be a set-multilinear polynomial of degree d, and $A \subseteq [d]$ be a set of indices. The **partial***

***derivative matrix* $M_A(f)$** *of f with respect to A is defined as follows: the rows are indexed by set-multilinear monomials g with respect to the partition $\bigsqcup_{i \notin A} X_i$ and the columns are indexed by set-multilinear monomials h with respect to the partition $\bigsqcup_{i \in A} X_i$. The value of $M_A(f)(g,h)$ is the coefficient of the monomial $g \cdot h$ in f.*

The notion of shape was defined by Arvind & Raja (2016), and it slightly differs from the non-commutative setting because we need to keep track of the indices of the variable sets given by the partition from which the variables belong. More precisely, given a partition of $X = X_1 \sqcup X_2 \sqcup \cdots \sqcup X_d$, we associate to any monomial $t \in \mathrm{Tree}(X)$ of degree d its ***shape*** $\mathrm{shape}(t) \in \mathrm{Tree}_d([d])$ defined as the tree obtained from t by replacing each label by its index in the partition. In particular if t is set-multilinear, then each element in $[d]$ appears exactly once as an index in $\mathrm{shape}(t)$. Hence we let $\mathcal{T}_d \subseteq \mathrm{Tree}_d([d])$ denote the set of trees such that all elements of $[d]$ appear exactly once as a label of a leaf.

Let $\mathcal{C}$ be a commutative circuit. We let $\tilde{f}$ denote the commutative non-associative polynomial computed by C when it is seen as non-associative. A ***parse tree*** of $\mathcal{C}$ is any shape $s \in \mathcal{T}_d$ for which there exists a monomial $t \in \mathrm{Tree}(X)$ whose coefficient in $\tilde{f}$ is nonzero and such that $s = \mathrm{shape}(t)$. Formally, we let

$$\mathrm{PT}(\mathcal{C}) = \left\{\mathrm{shape}(t) \mid \tilde{f}(t) \text{ non-zero}\right\} \cap \mathcal{T}_d.$$

REMARK 4.7. *Note that it may be the case that a circuit $\mathcal{C}$ computing a set-multilinear polynomial f computes a non-associative $\tilde{f}$ such that $\tilde{f}(t) \neq 0$ for some non set-multilinear monomials t, provided their sums collapse to 0 in the associative setting. We do not count such shapes as parse trees (this explains the intersection with $\mathcal{T}_d$ in the above definition), which leads to more general classes of circuits against which we shall obtain lower bounds.*

Given a shape $s \in \mathcal{T}_d$ and a node v of s, we let s_v denote the subtree rooted at v and $A_v \subseteq [d]$ denote the set of labels appearing on the leaves of s_v. We say that A_v is ***spanned*** by s.

Following the same roadmap as in the non-commutative setting we obtain the following counterpart of Theorem 4.1. We assume that the set of variables is partitioned into d parts of equal size n (this is a natural setting for polynomials such as the determinant, the permanent or the iterated matrix multiplication). In particular, it means that the polynomials we consider are of degree d and over nd variables.

THEOREM 4.8. *Let f be a set-multilinear polynomial computed by a circuit $\mathcal{C}$. Let $A \subseteq [d]$ and $\delta \in \mathbb{N}$ such that all parse trees of $\mathcal{C}$ span a subset at distance at most δ from A. Then $\mathcal{C}$ has size at least rank $(M_A(f))\, n^{-\delta} |PT(\mathcal{C})|^{-1}$.*

PROOF. As this proof is an adaptation of that of Theorem 4.1, we concentrate on highlighting the necessary changes.

Let $X = X_1 \sqcup X_2 \sqcup \cdots \sqcup X_d$ denote the underlying partition. Previously, we grouped together (sub-)trees and (sub-)contexts which correspond to a given interval of positions. In the commutative setting, we instead group together the (sub-)trees and (sub-)contexts which correspond to a given *subset* of positions, where a position is now being given by its index in the partition. Formally, for $A \subseteq [d]$, we let

$$\mathrm{Tree}_A(X) = \{t \in \mathrm{Tree}(X) \mid \text{ the set of indices of variables labeling } t \text{ is } A\},$$

and likewise,

$$\mathrm{Context}_A(X) = \{c \in \mathrm{Context}(X) \mid \text{ the set of indices of variables (different from } \square) \text{ labeling } c \text{ is } A\},$$

and finally

$$H_A = H_{|\mathrm{Tree}_A(X) \times \mathrm{Context}_{A^{\mathrm{C}}}(X)}.$$

Now, grouping together the columns of H_A which correspond to trees which have a given fixed shape s' (recall that a commutative shape contains the index in the partition of each leaf), and the rows which correspond to contexts which have a given fixed shape of context r yields the partial derivative matrix $M_A\left(f_{r[s']}\right)$, where the (commutative, associative) polynomial f_s is defined, for any commutative shape s, by

$$f_s(u) = \tilde{f}(s \text{ labelled by } u),$$

where the labeling respects the partition of X.

Hence, $\operatorname{rank}(H) \geq \operatorname{rank}(M_A(f_s))$ whenever A is spanned by s. The remainder of the proof exactly follows that of Theorem 4.1 and therefore we do not repeat it here. $\square$

A notable difference with the non-commutative setting is that now parse trees no longer span intervals of $[d]$ but subsets of $[d]$. As a consequence, if we follow the same technique as the one used to prove Theorem 4.2, we now groups together blocks corresponding to the same *subset* of $[d]$ and therefore the multiplicative factor is now 2^{-d} as there are 2^d such subsets. This yields the following counterpart for Theorem 4.2.

THEOREM 4.9. *Let f be a set-multilinear polynomial computed by a circuit $\mathcal{C}$. Let $A \subseteq [d]$ and $\delta \in \mathbb{N}$ such that all parse trees of $\mathcal{C}$ span a subset at distance at most δ from A. Then $\mathcal{C}$ has size at least $\operatorname{rank}(M_A(f))\, n^{-\delta} 2^{-d}$.*

PROOF. Again, we extend the ideas for the non-commutative setting to the commutative setting, and we reuse the notations of the proof of Theorem 4.2. As for proving Theorem 4.2, we mainly rely on a Lemma.

LEMMA 4.10. *Let $\tilde{f} : Tree(X) \to K$ be a non-associative commutative polynomial and let $\leq_{int}$ be a total order on subsets of $[d]$. For any commutative shape s, let $A(s)$ be the smallest (with respect to $\leq_{int}$) subset spanned by s. For any subset A, define a commutative associative polynomial by*

$$f_A(u) = \sum_{\substack{t \in Tree(X) \\ label(t) = u \\ A(shape(t)) = A}} \tilde{f}(t).$$

Then, one has $\max_A \operatorname{rank}(M_A(f_A)) \leq \operatorname{rank}\left(H_{\tilde{f}}\right)$.

The proof of Lemma 4.10 is very similar, yet a bit more pleasant than that of Lemma 4.4, since we no longer need to shift any interval. Formally, for $A \subseteq [d]$ we define

$$\mathscr{T}_A = \{t \in \mathrm{Tree}_A(X) \mid A \text{ is the smallest interval spanned by } \mathrm{shape}(t)\},$$

and likewise,

$$\mathscr{C}_A = \{c \in \mathrm{Context}_A(X) \mid A \text{ is the smallest interval spanned by } \mathrm{shape}(c)\}.$$

Now, the lemma follows from the fact that $M_A(f_A)$ is obtained by summing rows from $\mathscr{T}_A$ and columns from $\mathscr{C}_A$ in H.

The remainder of the proof is a very straightforward adaptation of the end of the proof of Theorem 4.2 from the non-commutative to the commutative setting. □

REMARK 4.11. *While in the non-commutative setting, Theorem 4.2 strengthens Theorem 4.1 (when d^2 is small), this is no longer the case in the commutative setting. Indeed, the maximal number of commutative parse trees being roughly $d!$ (the exact asymptotic is $\frac{\sqrt{2-\sqrt{2}}d^{d-1}}{e^d(\sqrt{2}-1)^{d+1}}$, see Sloane 2011), Theorem 4.8 and Theorem 4.9 are incomparable.*

5. Applications

In this section we instantiate our generic lower bound theorems on concrete classes of circuits. We first show how the weaker version (Theorem 4.1) yields the best lower bounds to date for skew and small non-skew depth circuits. Extending these ideas we obtain exponential lower bounds for $\left(\frac{1}{2}-\varepsilon\right)$-unbalanced circuits, an extension of skew circuits which are just slightly unbalanced. We also adapt the proof to ε-balanced circuits, which are slightly balanced. We then move on to our main results, which concern circuits with many parse trees, with lower bounds for both non-commutative and commutative settings.

Prior to that, we present a family of polynomials for which our lower bounds hold, and we state Lemma 5.1 which is used several times in our proofs.

High-ranked polynomials. The lower bounds we state below hold for any polynomial whose partial derivative matrices with respect to either a fixed subset A or all subsets have full rank. Such polynomials exist for all fields in both the commutative and non-commutative settings, and can be explicitly constructed. For instance the so-called Nisan-Wigderson polynomial (Kayal *et al.* 2014b)—inspired by the notion of designs by Nisan & Wigderson (1994)—has this property. In the commutative, set-multilinear setting, it is given by

$$NW_{n,d} = \sum_{\substack{h \in \mathbb{F}_n[z] \\ \deg(h) \leq d/2}} \prod_{i=1}^{d} x_{i,h(i)},$$

where $\mathbb{F}_n[z]$ denotes univariate polynomials with coefficients in the finite field of prime order n. In the non-commutative setting, we remove index i, and insist that the product $\prod_{i=1}^{d} x_{h(i)}$ is done along increasing values of i. The fact that there exists a unique polynomial $h \in \mathbb{F}_n[z]$ of degree at most $d/2$ which takes $d/2$ given values at $d/2$ given positions exactly implies that the partial derivative matrix of $NW_{n,d}$ with respect to any $A \subseteq [d]$ of size $d/2$ is a permutation matrix. This is then easily extended to any $A \subseteq [d]$.

A-balanced subsets. The following combinatorial Lemma is widely used to derive our lower bounds. Intuitively, a subset $B \subseteq [d]$ is far from a subset $A \subseteq [d]$ of size $d/2$ whenever it is A-balanced, meaning that $A \cap B$ and $A^{\mathrm{c}} \cap B$ have roughly the same size.

LEMMA 5.1. *Let $A, B \subseteq [d]$ be such that $|A| = d/2$. Then*

$$d(A, B) = d/2 - \big||A \cap B| - |A^{\mathrm{c}} \cap B|\big|.$$

PROOF. Let us first assume that $|B \cap A| \geq |B|/2$. This implies that $|\Delta(A,B)| \leq |\Delta(A^{c},B)|$, so

$$\begin{aligned}\operatorname{dist}(A,B) &= |\Delta(A,B)| \\ &= |A\cup B| - |A\cap B| \\ &= (|A| + |A^{c}\cap B|) - |A\cap B| \\ &= d/2 - (|A\cap B| - |A^{c}\cap B|) \\ &= d/2 - \big||A\cap B| - |A^{c}\cap B|\big|,\end{aligned}$$

where the last line also follows from the assumption that $|B\cap A| \geq |B|/2$. Now if $|B\cap A| < |B|/2$, it suffices to replace A with A^{c} in the previous proof to obtain the announced result. □

5.1. Applications in the non-commutative setting.

5.1.1. Skew, slightly unbalanced, slightly balanced and small non-skew depth circuits. We show how using Theorem 4.1 yields exponential lower bounds for four classes of circuits in the non-commutative setting. We adapt the ideas of Limaye *et al.* (2016) into our newly introduced vocabulary and easily obtain the same exponential lower bounds for skew circuits. Straightforward generalizations lead to previously unknown exponential lower bounds on slightly unbalanced and slightly balanced circuits. Finally, we also adapt (and shorten) their proof of a lower bound on small non-skew depth circuits. In each of these four cases the use of our weaker theorem, namely Theorem 4.1 suffices.

Skew Circuits A circuit $\mathcal{C}$ is ***skew*** if all its parse trees are skew, meaning that each node has at least one of its children which is a leaf. We let $I_{mid} = (d/4, 3d/4]$, which has size $d/2$. As a direct application of Theorem 4.1, we obtain the following result.

THEOREM 5.2. *Let f be a homogeneous non-commutative polynomial of degree d and on n variables such that $M_{I_{mid}}(f)$ has full rank $n^{d/2}$. Then any skew circuit computing f has size at least $2^{-d}n^{d/4}$.*

PROOF. The proof relies on the following two easy observations.

FACT 5.3. *Any skew shape spans intervals of each possible size, and in particular, an interval of size* $3d/4$.

PROOF. Let $s \in \mathrm{Tree}_d$ be a skew shape, v_1 be its root, and for all $i = 1 \ldots d-2$, v_{i+1} be the child of v_i which is not a leaf. Then any of the two children of v_{d-2} is a leaf, so it spans an interval of size 1. Now for each i, v_i spans an interval that includes $I_{v_{i+1}}$ and adds 1 to its size, so we easily conclude by induction. □

FACT 5.4. *Any interval of size* $3d/4$ *is at distance at most (in fact, equal to)* $d/4$ *from* I_{mid}.

PROOF. Indeed, let $I \subseteq [d]$ be an interval of size $3d/4$. Then $I_{mid} \subseteq I$ (see Figure 5.1). Hence by Lemma 5.1,

$$\begin{aligned} d(I, I_{mid}) &= d/2 - \big| |I \cap I_{mid}| - |I \cap I_{mid}^{\mathrm{C}}| \big| \\ &= d/2 - |d/2 - (|I| - d/2)| = d/4. \end{aligned}$$

□

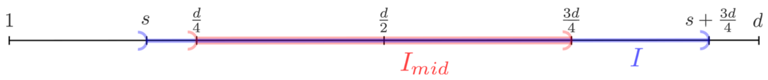


Figure 5.1: Any interval I of size $\frac{3d}{d}$ is at distance $\frac{d}{4}$ from I_{mid}.

A skew circuit has only skew parse trees, which all span an interval of size $3d/4$. Such an interval is at distance $d/4$ from I_{mid}, so the announced lower bound follows directly from Theorem 4.1, together with the fact that there are 2^d skew trees. □

REMARK 5.5. *Note that the factor* 2^{-d} *is easily replaced by* d^{-2} *by applying Theorem 4.2 instead, but we find it remarkable that simply using a decomposition of* H *into blocks is enough to obtain such an exponential lower bound.*

Slightly Unbalanced Circuits A circuit $\mathcal{C}$ computing a homogeneous non-commutative polynomial of degree d is said to be ***α-unbalanced*** if every multiplication gate has at least one of its children which computes a polynomial of degree at most αd.

THEOREM 5.6. *Let f be a homogeneous non-commutative polynomial of degree d and on n variables such that $M_{I_{mid}}(f)$ has full rank $n^{d/2}$. Then any $\left(\frac{1}{2}-\varepsilon\right)$-unbalanced circuit computing f has size at least $4^{-d}n^{\varepsilon d}$.*

This result improves over a previously known exponential lower bound on $\left(\frac{1}{5}\right)$-unbalanced circuits (Limaye *et al.* 2016).

PROOF. This is an adaptation of the proof of Theorem 5.2 about skew circuits. We now rely on these two observations, which respectively generalize Fact 5.3 and Fact 5.4:

FACT 5.7. *Any $(\frac{1}{2}-\varepsilon)$-unbalanced shape spans an interval of size between $3d/4-(\frac{1}{2}-\varepsilon)d/2$ and $3d/4+(\frac{1}{2}-\varepsilon)d/2$, that is, between $d/2+d\varepsilon/2$ and $d-d\varepsilon/2$.*

PROOF. Let α denote $(\frac{1}{2}-\varepsilon) < 1/2$ and let $s \in \mathrm{Tree}_d$ be an α-unbalanced shape of size d. We let v_1 be its root, and v_2 be the child of v_1 which spans the largest interval, which has size $I_{v_2} \geq (1-\alpha)d \geq \alpha d$. If both children of I_{v_2} span intervals of size $\leq \alpha d$, we set $r=2$, and otherwise iterate for $i=3\dots r$ until both children of v_r span intervals of size $\leq \alpha d$. Now, if we choose v_{r+1} to be a child of v_r, the cardinalities of the growing sequence of intervals $I_{v_{r+1}} \subseteq I_{v_r} \subseteq \cdots \subseteq I_{v_1} = [d]$ range from $\leq \alpha d$ to d with differences bounded by αd, so one of the interval has a size lying in $[3d/4-\alpha/2, 3d/4+\alpha/2]$. $\square$

FACT 5.8. *Any interval I of size $d/2+\varepsilon d/2 \leq |I| \leq d-\varepsilon d/2$ is at distance $\leq d/2-\varepsilon d/2$ from I_{mid}.*

PROOF. We make a case distinction and first assume that $I_{mid} \subseteq I$. Then, by Lemma 5.1, we have that

$$\begin{aligned}\mathrm{dist}(I, I_{mid}) &= d/2 - \big||I \cap I_{mid}| - |I \cap I^{\mathrm{C}}_{mid}|\big| \\ &= d/2 - |d/2 - (|I| - d/2)| \\ &= d - |I| < d/2 - \varepsilon d/2.\end{aligned}$$

Assume now that $I_{mid} \not\subseteq I$. Then, either $3d/4$ or $d/4+1$ does not belong to I. Both cases being symmetrical, we assume without loss of generality that $3d/4 \notin I$. We let $\ell = |I \cap I_{mid}|$. The current situation is depicted in Figure 5.2.

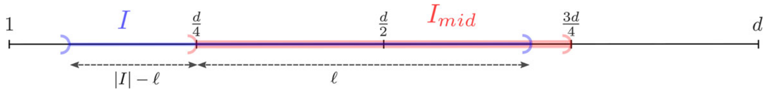


Figure 5.2: Illustrating I and I_{mid} when $3d/4 \notin I$.

It follows that $\big|I \cap]3d/4, d]\big| = 0$, and $\big|I \cap]1, d/4]\big| = (|I| - \ell) \leq d/4$. Multiplying this last inequality by two and summing with $-|I| \leq -d/2$ yields $|I| - 2\ell \leq 0$, so we obtain

$$\begin{aligned}\mathrm{dist}(I, I_{mid}) &= d/2 - \big||I \cap I_{mid}| - |I \cap I^{\mathrm{C}}_{mid}|\big| \\ &= d/2 - \big|\ell - (\big|I \cap [1, d/4]\big| + \big|I \cap]3d/4, d]\big|)\big| \\ &= d/2 - \big|\ell - (|I| - \ell)\big| \\ &= d/2 - (2\ell - |I|).\end{aligned}$$

Now since $|I| - \ell \leq d/4$, $-2\ell \leq -2|I| + d/2$, which leads to

$$\begin{aligned}\mathrm{dist}(I, I_{mid}) &= d/2 - 2\ell + |I| \\ &\leq d - |I| \leq d/2 - \varepsilon d/2,\end{aligned}$$

since $|I| \geq d/2 + \varepsilon d/2$. □

We conclude the proof by applying Theorem 4.1, just as we did for skew circuits. □

Slightly balanced circuits A circuit $\mathcal{C}$ computing a homogeneous non-commutative polynomial of degree d is said to be ***α-balanced*** if every multiplication gate which computes a polynomial of degree k has both of its children which compute polynomials of degree at least αk.

THEOREM 5.9. *Let f be a homogeneous non-commutative polynomial of degree d and on n variables such that $M_{[1,d/2]}(f)$ has full rank $n^{d/2}$. Then any ε-balanced circuit computing f has size at least $4^{-d}n^{\varepsilon d}$.*

PROOF. Let s be an ε-balanced shape, and r be the root of s. Let $I = [1, b]$ be the interval spanned by the left child of r. Since s is ε-balanced, $\varepsilon d \leq |I| = b \leq (1-\varepsilon)d$. Hence, I is at a distance of at most $d/2 - \varepsilon d$ from $[1, d/2]$, which allows us to conclude using Theorem 4.1. □

Note that is suffices to simply restrict the *last* multiplication in the circuit to be ε-balanced for the proof to carry on.

Small non-skew depth circuits A circuit $\mathcal{C}$ has ***non-skew depth k*** if all its parse trees are such that each path from the root to a leaf goes through at most k non-skew nodes, i.e., nodes for which the two children are inner nodes. We obtain an alternative proof of the exponential lower bound of Limaye *et al.* (2016) on non-skew depth k circuits as an application of Theorem 4.1. In the rest of this section we assume that $k \geq 30$, $p \geq 30$ is some multiple of 3 and $d = 12kp$. We will make extended use of the subset $A \subseteq [d]$ introduced in Limaye *et al.* (2016),

$$A = [1, 3kp] \cup \bigcup_{i=1}^{3k}]3(k+i)p + 2p, 3(k+i+1)p] \subseteq [d],$$

of size $6kp = d/2$ which is better understood in Figure 5.3.

THEOREM 5.10. *Let f be a homogeneous non-commutative polynomial of degree $d = 12kp$ and on n variables such that $M_A(f)$ has full rank $n^{d/2}$. Then any circuit of non-skew depth k computing f has size at least $4^{-d}n^{p/3} = 4^{-d}n^{d/36k}$.*

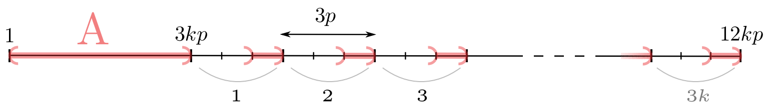


Figure 5.3: Subset $A \subseteq [d]$.

PROOF. We shall prove that any parse tree $s \in \mathrm{Tree}_d$ with non-skew depth k spans an interval $I(s)$ at distance $\leq d/2 - p/3$ from A. Then the result follows by applying Theorem 4.1.

Assume towards contradiction that a non-skew depth k shape $s \in \mathrm{Tree}_d$ spans only interval at distance $> d/2 - p/3$ from A. We consider (see Figure 5.4) the path $v_1 \cdots v_r$ in s from its root to the leaf with position $3kp$, and write u_i for $i \in r-1$, to refer to the child of v_i which is not v_{i+1}. Since s has non-skew depth k, at least $r-k$ nodes among $v_1, \ldots, v_{r-1}$ are leaves.

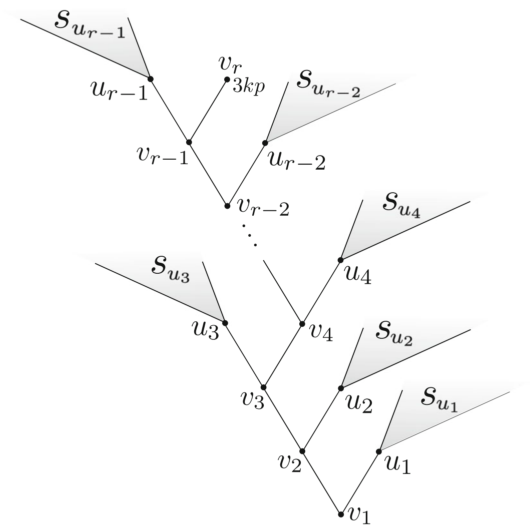


Figure 5.4: The path from the root v_1 to v_r, the leaf with position $3kp$.

We now state and prove some facts which then lead to a contradiction:

FACT 5.11. *For every $i \in [r]$, if u_i is the left child of v_i then $|I_{u_i}| < p/3$.*

PROOF. Indeed, u_i being at the left of the path to the leaf at position $3kp$, $I_{u_i} \subseteq [1, 3kp] \subseteq A$. But $\mathrm{dist}(I_{u_i}, A) > d/2 - p/3$, so it must be that $|I_{u_i}| < p/3$. □

FACT 5.12. *For every $i \in [r]$, if u_i is the right child of v_i then $|I_{u_i}| < 5p$.*

PROOF. Likewise, we now have $I_{u_i} \subseteq [3kp + 1, d]$. Intuitively, a large interval in this zone must contain roughly twice as much elements from A^{c} than from A, so they cannot be at distance close to the maximum $d/2$.

Let l be the number of blocks of the form $[3(k + i)p + 2p + 1, 3(k+i+1)p] \subseteq A$ which intersects I_{u_i}. By contradiction, assume that $|I_{u_i}| > 5p$. Note that it implies that $l \geq 2$.

Assume that $l = 2$, then $|A \cap I_{u_i}| \leq 2p$ hence, as $|I_{u_i}| \geq 5p$, $|A^{\mathrm{c}} \cap I_{u_i}| \geq 3p$. Therefore, using Lemma 5.1, $d(A, I_{u_i}) \leq d/2 - p \leq d_2 - p/3$.

Finally, assume that $l > 2$. Then $|A \cap I_{u_i}| \leq pl$ and $|A^{\mathrm{c}} \cap I_{u_i}| \geq 2p(l-1)$. Therefore, using Lemma 5.1, $d(A, I_{u_i}) \leq d/2 - pl + 2p \leq d/2 - p \leq d_2 - p/3$. □

FACT 5.13. *It must be that $r \geq 7kp$.*

PROOF. Indeed, since $[1, d] \setminus \{3kp\} = [1, 12kp] \setminus \{3kp\}$ is covered by the I_{u_i}, which have size bounded by $5p$ (thanks to Fact 5.11 and Fact 5.12) and among which all but k may have size > 1 (as we consider a circuit of non-skew depth k), there must be at least $12kp - 5kp = 7kp$ of them. □

FACT 5.14. *There is some index i_0 such that $u_{i_0}, u_{i_0+1}, \ldots, u_{i_0+20p/3-1}$ are all leaves in s.*

PROOF. Indeed, only k among the $7kp$ u_i's may not be leaves. By contradiction assume that all blocks of consecutive leaves have length smaller than $20p/3$, so overall length is $(20p/3)(k+1)+k < 7kp$ as we initially assumed that $k, p \geq 30$. This contradicts Fact 5.13. $\square$

We now consider the increasing sequence of intervals $I_{v_{i_0+20p/3-1}} \subseteq I_{v_{i_0+20p/3-2}} \subseteq \cdots \subseteq I_{v_{i_0}}$ (where the nodes $u_{i_0}, u_{i_0+1}, \ldots, u_{i_0+20p/3-1}$ are those given by Fact 5.14), which we simply denote $I_1 \subseteq I_2 \subseteq \cdots \subseteq I_{20p/3}$. Each $I_i = [a_i, b_i]$ contains $3kp$, and $|I_{i+1}| = |I_i| + 1$. We let $n_i = |I_i \cap A|$ and $m_i = |I_i \cap A^c|$. The assumption $d(A, I_i) > d/2 - p/3$ can be rephrased, thanks to Lemma 5.1, as $|n_i - m_i| \leq p/3$.

First, note that for all $j < 6p$, $b_j \notin \{3(k+i)p + 2p + 1 \mid 1 \leq i \leq 3k\}$. Indeed, for such a j one would have $|n_{j+2p/3+1} - m_{j+2p/3+1}| = |n_j - m_j| + 2p/3 + 1 > p/3$ leading to a contradiction. Therefore, all the b_j for $j = 1, \ldots, 6p-1$ belong to $[3(k+i)p + 2p + 2, 3(k+i+1)p + 2p]$ for some $1 \leq i \leq 3k$. Hence, $m_{6p-1} - m_1 \leq 2p$, which implies that $n_{6p-1} - n_1 \geq 4p$.

Finally,

$$\begin{aligned}
2p/3 &\geq \big||n_1 - m_1| - |n_{6p-1} - m_{6p-1}|\big| \\
&\geq |n_{6p-1} - m_{6p-1}| - |n_1 - m_1| \\
&\geq n_{6p-1} - m_{6p-1} + m_1 - n_1 \\
&\geq 4p - 2p
\end{aligned}$$

which leads to a contradiction and concludes the proof. $\square$

5.1.2. Circuits with many parse trees. We now turn our focus to ***k-PT circuits*** which are circuits with at most k different parse trees. We first start by a key technical lemma that works both in the non-commutative and commutative (later discussed in Section 5.2) settings.

Balanced subsets For $s \in \mathrm{Tree}_d$ and $X \subseteq [d]$, we define

$$\mathrm{dist}(X, s) = \min\{\mathrm{dist}(X, A) \mid A \text{ spanned by } s\}.$$

In the following, we let $\binom{[d]}{d/2}$ denote the subsets of $[d]$ of size $d/2$. For a subset $\mathcal{P} \subseteq 2^{[d]}$ we write $\mathcal{U}(\mathcal{P})$ for the uniform distribution over $\mathcal{P}$.

Recall that, following Lemma 5.1, if $X \in \binom{[d]}{d/2}$ and $A \subseteq [d]$, $\mathrm{dist}(X, A) > d/2 - \delta$ rewrites as $\big||A \cap X| - |A^{\mathrm{c}} \cap X|\big| \leq \delta$, meaning that A is X-balanced.

The following lemma is a subtle probabilistic analysis bounding the number of subsets that are balanced over all subsets spanned by a given fixed shape s. This will later entail the existence of a subset which is close to all parses trees in $\mathrm{PT}(\mathcal{C})$, provided $|\mathrm{PT}(\mathcal{C})|$ is not too large. It holds in both the non-commutative (in which it was originally proved) and the commutative settings.

Lemma 5.15 (Adapted from Claim 15 in Lagarde *et al.* 2018). *Let $s \in Tree_d$ be a shape with d leaves, and $\delta \leq \sqrt{d}/2$. Then*

$$\Pr_{X \sim \mathcal{U}\left(\binom{[d]}{d/2}\right)} \left[dist(X, s) > d/2 - \delta\right] \leq 2^{-\alpha d/\delta^2},$$

where α is some positive constant.

We shall use an intermediate result from the aforementioned paper. Their proof (based on a greedy construction) can be read just as such in the commutative setting.

Lemma 5.16 (Subclaim 21 in Lagarde *et al.* 2018). *Let $s \in Tree_d$, and r, t be integers such that $rt \leq d/4$. Then there exists a sequence $u_1, \ldots, u_r$ of nodes of s such that for all $i \in [r]$,*

$$\left| A_{v_i} \setminus \left(\bigcup_{j=1}^{i-1} A_{u_j} \right) \right| \geq t.$$

We now give the proof of Lemma 5.16 in the commutative setting, noting that the proof in the non-commutative setting is a restricted version of the one we give, where spanned subsets of $[d]$ are replaced by spanned intervals.

Proof. We pick $t = \delta^2$ and $r = \frac{d}{4\delta^2}$, and apply Lemma 5.16 to obtain a sequence $v_1, ..., v_r$ of nodes of s. Then we have:

$$\begin{aligned}
&\Pr_{X\sim\mathcal{U}\left(\binom{[d]}{d/2}\right)}\left[\mathrm{dist}(X,s) > d/2-\delta\right]\\
&\quad = \Pr_{X\sim\mathcal{U}\left(\binom{[d]}{d/2}\right)}\left[\text{for all node } v \text{ of } s, \left||X\cap A_v| - |X^{\mathrm{c}}\cap A_v|\right| \le \delta\right]\\
&\quad \le d \Pr_{X\sim\mathcal{U}\left(2^{[d]}\right)}\left[\text{for all node } v \text{ of } s, \left||X\cap A_v| - |X^{\mathrm{c}}\cap A_v|\right| \le \delta\right]
\end{aligned}$$

The last inequality follows from the general fact, applied using $2^d \le d\binom{d}{d/2}$, that, for any event E and finite subsets $\mathcal{P} \subseteq \mathcal{P}'$ with $|\mathcal{P}'| \le k|\mathcal{P}|$ one has

$$\Pr_{A\sim\mathcal{U}(\mathcal{P})}(E) \le k \Pr_{A\sim\mathcal{U}(\mathcal{P}')}(E).$$

Following from there, we let E_i, for $i \in [r]$, be the event $\left||X \cap A_{v_i}| - |X^{\mathrm{c}} \cap A_{v_i}|\right| \le \delta$, and obtain

$$\begin{aligned}
&d \Pr_{X\sim\mathcal{U}\left(2^{[d]}\right)}\left[\text{for all node } v \text{ of } s, \left||X\cap A_v| - |X^{\mathrm{c}}\cap A_v|\right| \le \delta\right]\\
&\quad \le d \Pr_{X\sim\mathcal{U}\left(2^{[d]}\right)}\left[\forall i\in[r], E_i\right]\\
&\quad \le d \prod_{i=1}^{r} \Pr_{X\sim\mathcal{U}\left(2^{[d]}\right)}\left[E_i \mid \forall j<i, E_j\right]
\end{aligned}$$

In order to bound the terms $\Pr_{X\sim\mathcal{U}\left(2^{[d]}\right)}[E_i \mid \forall j < i, E_j]$ we use the following consequence of the Central Limit theorem.

FACT 5.17. *There exist $\beta < 1$ such that for all random variable Y following an unbiased binomial law of parameter n, and all interval I with $|I| \le 2\sqrt{n}$, one has $\Pr(Y \in I) \le \beta$.*

If X is sampled uniformly among $[d]$ and $X \cap \left(\bigcup_{j<i} A_{u_j}\right)$ is fixed, let $e = |X \cap \left(\bigcup_{j<i} A_{u_j}\right)| - |X^{\mathrm{c}} \cap \left(\bigcup_{j<i} A_{u_j}\right)|$. Then the event E_i can be rephrased as having a random variable following an unbiased binomial law of parameter $t = \delta^2$ sit in $[-\delta - e, \delta - e]$ of size 2δ, which is bounded by β thanks to Fact 5.17. Hence,

$$\Pr_{X\sim\mathcal{U}\left(\binom{[d]}{d/2}\right)}\left[\mathrm{dist}(X,s) > d/2-\delta\right] w \le d\beta^r = d\beta^{\frac{d}{4\delta^2}} \le 2^{-\alpha d/\delta^2}$$

for some positive constant α. □

Superpolynomial lower bounds Lagarde *et al.* (2018) obtained a superpolynomial lower bound for superpolynomial k (up to $k = 2^{d^{\frac{1}{3}-\varepsilon}}$). In the statement below, the first item shows how to obtain the same result using Theorem 4.1, while the second item improves the previous bound by applying Theorem 4.2 instead.

THEOREM 5.18. *Let f be a homogeneous non-commutative polynomial of degree d and with n variables such that, for all $A \subseteq [d]$, $M_A(f)$ has full rank. Let $\varepsilon > 0$. Then for large enough d,*

(i) any $2^{d^{1/3-\varepsilon}}$-PT circuit computing f has size at least $2^{d^{1/3}(\log n - d^{-\varepsilon})}$;

(ii) any $2^{d^{1-\varepsilon}}$-PT circuit computing f has size at least $n^{d^{\varepsilon/3}} d^{-2}$.

PROOF. Let $\mathcal{C}$ be a k-PT circuit computing f, and $\delta \leq \sqrt{d}$. We first show that there exists a subset $A \subseteq [d]$ which is close to all parse trees in $\mathcal{C}$. Indeed, a union bound and Lemma 5.15 yield

$$\Pr_{A \sim \mathcal{U}\left(\binom{[d]}{d/2}\right)} \left[\exists s \in \mathrm{PT}(\mathcal{C}), \mathrm{dist}(A, s) > d/2 - \delta\right] \leq \sum_{s \in \mathrm{PT}(\mathcal{C})} \Pr_{A \sim \mathcal{U}\left(\binom{[d]}{d/2}\right)} \left[\mathrm{dist}(A, s) > d/2 - \delta\right] \leq k 2^{-\alpha d/\delta^2}$$

for large enough d.

Choosing appropriate values for δ and k and applying Theorem 4.1 (*resp.* Theorem 4.2) leads the first (*resp.* second) item.

(i) Choosing $\delta = d^{1/3}$ and $k = 2^{d^{1/3-\varepsilon}}$, we have that $k 2^{-\alpha d/\delta^2} = 2^{d^{1/3-\varepsilon} - \alpha d^{1/3}} < 1$, This implies the existence of a subset $A \subseteq [d]$ of size $d/2$ such that for all $s \in \mathrm{PT}(\mathcal{C}), \mathrm{dist}(A, s) \leq d/2 - \delta$, that is, any $s \in \mathrm{PT}(\mathcal{C})$ spans an interval $I(s)$ at distance at most $d/2 - \delta$ from A. Finally, we apply Theorem 4.1 to obtain

$$|\mathcal{C}| \geq \mathrm{rank}(M_A(f))\, n^{-(d/2-\delta)} k^{-1} = n^{d/2} n^{-(d/2 - d^{1/3})} 2^{-d^{1/3-\varepsilon}} = 2^{d^{1/3}(\log n - d^{-\varepsilon})}.$$

(ii) Choosing $\delta = d^{\varepsilon/3}$ and $k = 2^{d^{1-\varepsilon}}$, we have that $k2^{-\alpha d/\delta^2} = 2^{d^{1-\varepsilon} - \alpha d^{1-\frac{2}{3}\varepsilon}} < 1$, which again lets us choose $A \subseteq [d]$ of size $d/2$ and such that for all $s \in \mathrm{PT}(\mathcal{C})$, $\mathrm{dist}(s, A) \leq d/2 - \delta$. Now, applying Theorem 4.2 we obtain

$$|\mathcal{C}| \geq \mathrm{rank}\left(M_A\left(f\right)\right) n^{-(d/2-\delta)} d^{-2} = n^{\delta} d^{-2} = n^{d^{\varepsilon/3}} d^{-2}.$$

□

In the second item, the bound $2^{d^{1-\varepsilon}}$ on the number of parse trees is to be compared to the total number of shapes of size d which is bounded by 2^{2d} as noticed in Remark 3.2. As explained in the introduction this means that we obtain superpolynomial lower bounds for any class of circuits which has a small defect in the exponent of the total number of parse trees.

5.2. Applications in the commutative setting. Regarding application in the commutative setting, we again consider the class of k-PT circuits which are set-multilinear circuits with at most k different commutative parse trees. Recall from Section 4.3 that in the commutative set-multilinear setting, parse trees are shapes whose leaves are labelled by integers without repetition. In particular the number of parse trees is roughly bounded by $d!$ (see Remark 4.11).

Arvind & Raja (2016) showed a superpolynomial lower bound for k-PT circuits computing set-multilinear polynomial for sublinear k (up to $k = d^{1/2-\varepsilon}$).

We improve this to superpolynomial k (up to $k = 2^{d^{1-\varepsilon}}$).

However, the generic lower bound theorems, namely Theorem 4.8 and Theorem 4.9, are not exactly the same, so we obtain two incomparable bounds. In the following lower bounds, the set-multilinear polynomials that we consider have their variables partitionned into $X = X_1 \sqcup X_2 \sqcup \cdots \sqcup X_d$ with $n = |X_i|$ for all i.

THEOREM 5.19. *Let f be a set-multilinear commutative polynomial such that for all $A \subseteq [d]$, the matrix $M_A\left(f\right)$ has full rank. Let $\varepsilon > 0$. Then for large enough d,*

(i) any $2^{d^{1/3-\varepsilon}}$-PT circuit computing f has size at least $2^{d^{1/3}(\log n - d^{-\varepsilon})}$;

(ii) any $2^{d^{1-\varepsilon}}$-PT circuit computing f has size at least $n^{d^{\varepsilon/3}}d^{-2}$. In particular, this lower bound is super polynomial when d is at most a polynomial in $\log n$.

Proof. Let $\mathcal{C}$ be a k-PT circuit computing f, and $\delta \leq \sqrt{d}$. By union bound and Lemma 5.15 for the commutative setting,

$$\Pr_{A\sim\mathcal{U}\left(\binom{[d]}{d/2}\right)}\left[\exists s \in \mathrm{PT}\left(\mathcal{C}\right), \mathrm{dist}(A,s) > d/2-\delta\right]$$
$$\leq \sum_{s\in\mathrm{PT}(\mathcal{C})}\Pr_{A\sim\mathcal{U}\left(\binom{[d]}{d/2}\right)}\left[\mathrm{dist}(A,s) > d/2-\delta\right] \leq k2^{-\alpha d/\delta^2}$$

Choosing appropriate values for δ and k and applying Theorem 4.8 (*resp.* Theorem 4.9) leads the first (*resp.* second) item.

(i) Choosing $\delta = d^{1/3}$ and $k = 2^{d^{1/3-\varepsilon}}$, we have that $k2^{-\alpha d/\delta^2} = 2^{d^{1/3-\varepsilon}-\alpha d^{1/3}} < 1$. Hence, picking a subset $A \subseteq [d]$ of size $d/2$ such that any $s \in \mathrm{PT}\left(\mathcal{C}\right)$ spans an interval $I(s)$ at distance at most $d/2-\delta$ from A, and applying Theorem 4.8 yields

$$\begin{aligned}|\mathcal{C}| \geq \mathrm{rank}\left(M_A\left(f\right)\right)n^{-(d/2-\delta)}k^{-1} &= n^{d/2}n^{-(d/2-d^{1/3})}2^{-d^{1/3-\varepsilon}}\\ &= 2^{d^{1/3}(\log n - d^{-\varepsilon})}.\end{aligned}$$

(ii) Choosing $\delta = d^{\varepsilon/3}$ and $k = 2^{d^{1-\varepsilon}}$, we have that $k2^{-\alpha d/\delta^2} = 2^{d^{1-\varepsilon}-\alpha d^{1-\frac{2}{3}\varepsilon}} < 1$. Hence, picking a subset $A \subseteq [d]$ of size $d/2$ and such that for all $s \in \mathrm{PT}\left(\mathcal{C}\right)$, $\mathrm{dist}(s,A) \leq d/2-\delta$, and applying Theorem 4.9 yields

$$|\mathcal{C}| \geq \mathrm{rank}\left(M_A\left(f\right)\right)n^{-(d/2-\delta)}k^{-1} = n^{\delta}2^{-d^{1-\varepsilon}} = n^{d^{\varepsilon/3}}2^{-d}.$$

□

6. Discussion

We presented a new tool for proving lower bounds for arithmetic circuits in the form of the Hankel matrix. We obtained strong lower bounds both in the commutative and non-commutative settings using generic decompositions of the Hankel matrix. A natural question is how far this approach can be pushed. The first remark is that the rank of the Hankel matrix is exactly the size of the smallest circuit computing a given (non-associative) polynomial, hence the potential loss can only be in analyzing the Hankel matrix. Limaye *et al.* (2016) defined a polynomial computed by a circuit of polynomial size but such that all partial derivative matrices have full rank: this shows that one cannot use our decomposition of the Hankel matrix to obtain strong lower bounds for the class of all circuits. This limitation is an invitation to get a deeper understanding of the Hankel matrix and to find other ways of decomposing it.

On a different perspective, the Hankel matrix has been successfully used as a data structure for learning algorithms (in both supervised and unsupervised settings). It is tempting, using the characterization that we present in this paper, to construct algorithms for learning polynomials relying on the Hankel matrix as algorithmic representation.

References

Manindra Agrawal, Rohit Gurjar, Arpita Korwar & Nitin Saxena (2015). Hitting-Sets for ROABP and Sum of Set-Multilinear Circuits. *SIAM Journal on Computing* **44**(3), 669–697.

Eric Allender, Jia Jiao, Meena Mahajan & V. Vinay (1998). Non-Commutative Arithmetic Circuits: Depth Reduction and Size Lower Bounds. *Theoretical Computer Science* **209**(1-2), 47–86.

Vikraman Arvind, Rajit Datta, Partha Mukhopadhyay & S. Raja (2017). Efficient Identity Testing and Polynomial Factorization in Nonassociative Free Rings. In *Proceedings of the 42nd International Symposium on Mathematical Foundations of Computer Science (MFCS 2017)*, volume 83 of *LIPIcs*, 38:1–38:13. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-046-0. ISSN 1868-8969.

Vikraman Arvind & S. Raja (2016). Some Lower Bound Results for Set-Multilinear Arithmetic Computations. *Chicago Journal of Theoretical Computer Science* **2016**.

Walter Baur & Volker Strassen (1983). The Complexity of Partial Derivatives. *Theoretical Computer Science* **22**, 317–330.

Symeon Bozapalidis & Olympia Louscou-Bozapalidou (1983). The Rank of a Formal Tree Power Series. *Theoretical Computer Science* **27**, 211–215.

Marco L. Carmosino, Russell Impagliazzo, Shachar Lovett & Ivan Mihajlin (2018). Hardness Amplification for Non-Commutative Arithmetic Circuits. In *Proceedings of the 33rd Computational Complexity Conference (CCC 2018)*, volume 102 of *LIPIcs*, 12:1–12:16. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik.

Suryajith Chillara & Partha Mukhopadhyay (2019). Depth-4 Lower Bounds, Determinantal Complexity: A Unified Approach. *Computational Complexity* **28**(4), 545–572.

Zeev Dvir, Guillaume Malod, Sylvain Perifel & Amir Yehudayoff (2012). Separating multilinear branching programs and formulas. In *Proceedings of the 44th Symposium on Theory of Computing Conference (STOC 2012)*, 615–624. ACM.

Nathanaël Fijalkow, Guillaume Lagarde & Pierre Ohlmann (2018). Tight Bounds using Hankel Matrix for Arithmetic Circuits with Unique Parse Trees. *Electronic Colloquium on Computational Complexity (ECCC)* **25**, 38. URL https://eccc.weizmann.ac.il/report/2018/038.

Michel Fliess (1974). Matrices de Hankel. *Journal de Mathématiques Pures et Appliquées* **53**, 197–222.

Michael A. Forbes, Ramprasad Saptharishi & Amir Shpilka (2014). Hitting sets for multilinear read-once algebraic branching programs, in any order. In *Proceedings of the 46th Symposium on Theory of Computing, (STOC 2014)*, 867–875. ACM.

Hervé Fournier, Nutan Limaye, Guillaume Malod & Srikanth Srinivasan (2014). Lower bounds for depth 4 formulas computing iterated matrix multiplication. In *Proceedings of the 46th Symposium on Theory of Computing (STOC 2014)*, 128–135. ACM.

Ankit Gupta, Pritish Kamath, Neeraj Kayal & Ramprasad Saptharishi (2014). Approaching the Chasm at Depth Four. *J. ACM* **61**(6), 33:1–33:16. URL https://doi.org/10.1145/2629541.

Rohit Gurjar, Arpita Korwar, Nitin Saxena & Thomas Thierauf (2017). Deterministic Identity Testing for Sum of Read-Once Oblivious Arithmetic Branching Programs. *Computational Complexity* **26**(4), 835–880.

Pavel Hrubeš, Avi Wigderson & Amir Yehudayoff (2010). Relationless Completeness and Separations. In *Proceedings of the 25th Annual IEEE Conference on Computational Complexity (CCC 2010)*, 280–290. IEEE Computer Society.

Pavel Hrubeš, Avi Wigderson & Amir Yehudayoff (2011). Non-commutative circuits and the sum-of-squares problem. *Journal of the American Mathematical Society* **24**(3), 871–898.

Laurent Hyafil (1977). The Power of Commutativity. In *Proceedings of the 18th Annual Symposium on Foundations of Computer Science (FOCS 1977)*, 171–174. IEEE Computer Society.

Mark Jerrum & Marc Snir (1982). Some Exact Complexity Results for Straight-Line Computations over Semirings. *J. ACM* **29**(3), 874–897. URL https://doi.org/10.1145/322326.322341.

Valentine Kabanets & Russell Impagliazzo (2003). Derandomizing polynomial identity tests means proving circuit lower bounds. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC 2003)*, 355–364. ACM.

Neeraj Kayal, Nutan Limaye, Chandan Saha & Srikanth Srinivasan (2014a). An Exponential Lower Bound for Homogeneous Depth Four Arithmetic Formulas. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, 61–70. IEEE Computer Society. URL https://doi.org/10.1109/FOCS.2014.15.

NEERAJ KAYAL, CHANDAN SAHA & RAMPRASAD SAPTHARISHI (2014b). A super-polynomial lower bound for regular arithmetic formulas. In *Proceedings of the 46th Symposium on Theory of Computing, (STOC 2014)*, 146–153. ACM.

MRINAL KUMAR & SHUBHANGI SARAF (2017). On the Power of Homogeneous Depth 4 Arithmetic Circuits. *SIAM J. Comput.* **46**(1), 336–387. URL https://doi.org/10.1137/140999335.

GUILLAUME LAGARDE, NUTAN LIMAYE & SRIKANTH SRINIVASAN (2018). Lower Bounds and PIT for Non-commutative Arithmetic Circuits with Restricted Parse Trees. *Computational Complexity* 1–72.

GUILLAUME LAGARDE, GUILLAUME MALOD & SYLVAIN PERIFEL (2016). Non-commutative computations: lower bounds and polynomial identity testing. *Electronic Colloquium on Computational Complexity (ECCC)* **23**, 94.

NUTAN LIMAYE, GUILLAUME MALOD & SRIKANTH SRINIVASAN (2016). Lower Bounds for Non-Commutative Skew Circuits. *Theory of Computing* **12**(1), 1–38.

GUILLAUME MALOD & NATACHA PORTIER (2008). Characterizing Valiant's algebraic complexity classes. *Journal of Complexity* **24**(1), 16–38.

NOAM NISAN (1991). Lower Bounds for Non-Commutative Computation (Extended Abstract). In *Proceedings of the 23rd Symposium on Theory of Computing (STOC 1991)*, 410–418. ACM.

NOAM NISAN & AVI WIGDERSON (1994). Hardness vs Randomness. *Journal of Computer and System Sciences* **49**(2), 149–167.

NOAM NISAN & AVI WIGDERSON (1997). Lower Bounds on Arithmetic Circuits Via Partial Derivatives. *Computational Complexity* **6**(3), 217–234.

C. RAMYA & B. V. RAGHAVENDRA RAO (2018). Lower Bounds for Special Cases of Syntactic Multilinear ABPs. In *Proceedings of the 24th International Computing and Combinatorics Conference (COCOON 2018)*, volume 10976 of *Lecture Notes in Computer Science*, 701–712. Springer.

Ran Raz & Amir Shpilka (2005). Deterministic polynomial identity testing in non-commutative models. *Computational Complexity* **14**(1), 1–19.

Ramprasad Saptharishi & Anamay Tengse (2017). Quasi-polynomial Hitting Sets for Circuits with Restricted Parse Trees. *Electronic Colloquium on Computational Complexity (ECCC)* **24**, 135.

N. J. A. Sloane (editor) (2011). *The On-Line Encyclopedia of Integer Sequences*. Number of labeled rooted unordered binary trees (each node has out-degree ≤ 2), https://oeis.org/A036774.

Seinosuke Toda (1992). Classes of Arithmetic Circuits Capturing the Complexity of Computing the Determinant. *IEICE Trans. Inf. Systems* **E75-D**(1), 116–124.

Leslie G. Valiant (1979). The Complexity of Computing the Permanent. *Theoretical Computer Science* **8**, 189–201.

Manuscript received 23 March 2020

Nathanaël Fijalkow
CNRS, LaBRI, Bordeaux, France
The Alan Turing Institute of data science, London,
United Kingdom
Nathanael.Fijalkow@labri.fr

Guillaume Lagarde
LaBRI, Bordeaux, France
Guillaume.Lagarde@labri.fr

Pierre Ohlmann
Université de Paris, IRIF, CNRS, F-75013 Paris, France
Pierre.Ohlmann@irif.fr

Olivier Serre
Université de Paris, IRIF, CNRS, F-75013 Paris, France
Olivier.Serre@cnrs.fr