Lower Bounds for Constant Depth Algebraic Circuits

A thesis submitted to

Chennai Mathematical Institute

in Partial Fulfillment of the Requirements for the Degree of

Master of Science

in

Computer Science

by

Sagnik Dutta

(MCS202112)

under the supervision of

Prof. Nitin Saxena

Department of Computer Science

Chennai Mathematical Institute

May, 2023

Declaration

I hereby declare that this thesis represents my own work done under the guidance of my supervisor. It has not been submitted anywhere else for a degree or a diploma.

I have complied with the norms and research ethics guidelines of the University. Further, appropriate credit has been given within this thesis where reference has been made to the work of others.

Sagnik Dutta

Date: May 31, 2023

(Sagnik Dutta)
(MCS202112)

CERTIFICATE

This is to certify that the project report entitled "**Lower Bounds for Constant Depth Algebraic Circuit**" submitted by **Sagnik Dutta** (Roll No. MCS202112) to Chennai Mathematical Institute towards partial fulfillment of requirements for the degree of Master of Science in Computer Science is a record of bona fide work carried out by him under my supervision guidance during Sep'21-May'22.

Date: May 31, 2023

Prof. Nitin Saxena
Dept. of Comp. Sci. & Engg.
IIT Kanpur

Abstract

Name of the student: **Sagnik Dutta** Roll No: **MCS202112**

Thesis title: **Lower Bounds for Constant Depth Algebraic Circuits**

Degree for which submitted: **Master of Science**

Department: **Computer Science**

Thesis supervisor: **Prof. Nitin Saxena**

An arithmetic circuit is a natural model for computing polynomials over a field $\mathbb{F}$. It is a directed acyclic graph whose leaves are input variables $x_1, \cdots, x_n$ and constants from the field $\mathbb{F}$. The internal nodes are addition or multiplication gates. The *size* of a circuit is the number of edges in it and the *depth* is the length of the longest directed path in it. Given a partition of the variable set $\{x_1, \cdots, x_n\}$ into sets $X_1, \cdots, X_d$, a polynomial is called *set-multilinear* with respect to this partition if every monomial of the polynomial contains exactly one variable from each set X_i. If every node of a circuit computes a set-multilinear polynomial, then it is called a *set-multilinear circuit.*

The main goal of Algebraic Complexity Theory is to exhibit an explicit polynomial to compute which circuits of superpolynomial size are required. By an explicit polynomial, we mean a polynomial where given the exponent vector of a monomial, we can compute the coefficient of this monomial in the polynomial efficiently. But some interesting *depth reduction* results show that strong enough lower bounds for constant depth circuits yield superpolynomial lower bounds for general algebraic

circuits. Hence, our motivation is to find strong lower bounds for constant depth algebraic circuits.

In a recent breakthrough result [LST], the first-ever superpolynomial lower bounds on the size of constant depth algebraic circuits were shown. The main idea of the paper was to first convert general algebraic circuits to set-multilinear circuits without much blowup in depth and size. Thus, strong enough lower bounds on set-multilinear constant depth circuits would imply constant depth general circuit lower bounds. The strong set-multilinear lower bound was achieved by considering a partition of the variables into sets of different sizes and using this discrepancy of set sizes crucially.

In this thesis, we improve the lower bounds in [LST]. The strategy we employed is to pick the set sizes more carefully. We design a number-theoretic algorithm to give this better choice of the set sizes depending on the depth we are working with and this lets us prove a stronger lower bound.

Acknowledgements

I am deeply indebted to many people who have made this thesis possible with their generous support and encouragement.

First and foremost, I would like to express my sincere gratitude to Prof. Nitin Saxena for his guidance throughout this project. I am extremely thankful for all the long discussions where he always had invaluable insights to offer. His research methods and ways of tackling a problem have always amazed me. I hope to emulate his enthusiasm and dedication towards research in my future endeavors.

I would like to thank Bhargav, for being an excellent collaborator on this project. The discussions with him always sparked new ideas and helped clear my doubts. Additionally, I want to thank my labmates Sanyam, Sayak, Diptajit and Devansh for all the joyful moments we shared. They made the 10-month stay at IIT Kanpur an unforgettable experience.

I am immensely grateful to the faculty of CMI for providing me with a strong foundation in computer science through the excellent courses they offer. I would also like to thank all my friends at CMI, Kolkata and Berhampore for their company and support. Life would not be fun without their presence.

Finally, I would like to express my deepest gratitude to my family. Their love and unwavering support through the years have been the main driving force behind my pursuit of knowledge. I dedicate this thesis to them.

Publication

This thesis is based on the work in

[BDS22] **Improved Lower Bound, and Proof Barrier, for Constant Depth Algebraic Circuits**

C. S. Bhargav, Sagnik Dutta and Nitin Saxena

MFCS 2022

Contents

Declaration **i**

Certificate **ii**

Abstract **iii**

Acknowledgements **v**

Contents **vii**

1 Introduction **1**
- 1.1 Our Models of Computation 1
- 1.2 VP and VNP: Algebraic Complexity Classes 3
- 1.3 Before 2021: Lower Bounds for Constant Depth Circuits 4
- 1.4 2021: The LST Breakthrough 4
- 1.5 Some More Recent Works 5
- 1.6 Contribution of this Thesis 6

2 Preliminaries **8**
- 2.1 Words 8
- 2.2 Relative Rank: The Complexity Measure 9
- 2.3 Word Polynomials 10

3 Lower bound proof overview **12**
- 3.1 Proof overview of Theorem 1.2 for $\Delta = 3$ 13
- 3.2 Obstacles in extending the above proof strategy to product-depth 4 and how to overcome them 15

4 Improved lower bound for constant depth circuits **16**
- 4.1 Proof of the Lower Bounds 16
- 4.2 Existence of (d, Δ)-nice tuples 21
  - 4.2.1 Defining the sequences $\{b_m\}$, $\{c_m\}$ and the tuple (p, q): 22
  - 4.2.2 Bounds on the values of b_m and $|c_m|$ 23
  - 4.2.3 Missing proofs of technical lemmas 27

Bibliography **32**

Chapter 1

Introduction

1.1 Our Models of Computation

Fix an underlying field $\mathbb{F}$.

Definition 1.1: Arithmetic Circuits and Formulas

An **arithmetic circuit** is a directed acyclic graph with one sink (vertex with zero outdegree) called the output gate. The leaves are labelled by variables $x_1, \cdots, x_n$ or elements from $\mathbb{F}$. The internal nodes are either addition $(+)$ or multiplication $(\times)$ gates. Each node of the circuit naturally computes a polynomial in $\mathbb{F}(x_1, \cdots, x_n)$. The circuit is said to compute a polynomial f if the output gate computes the polynomial f.

An **arithmetic formula** is a circuit whose every internal node has outdegree at most 1.

Without loss of generality, we can assume that the circuit or formula has alternating layers of addition and multiplication gates, with edges going only from one layer to the next layer.

There are some interesting complexity measures associated with circuits or formulas:

- **Size:** the total number of nodes and edges in the circuit.
- **Depth:** the number of layers in the circuit.
- **Product-depth:** the number of layers of multiplication gates in the circuit.

Definition 1.2: Set-multilinear polynomials and circuits

Let the underlying variable set $\{x_1, \cdots, x_n\}$ be partitioned into d sets $X_1, \cdots, X_d$. Then, a polynomial $f \in \mathbb{F}(x_1, \cdots, x_n)$ is said to be **set-multilinear** with respect to this partition if every monomial of it contains one variable from each variable set X_i.

If every node of a circuit computes a set-multilinear polynomial, then it is called a **set-multilinear circuit**.

An example of a set-multilinear polynomial is the **Iterated Matrix Multiplication Polynomial** $\text{IMM}_{n,d}$ which is defined on nd^2 variables. The variables are partitioned into d sets $X_1, \cdots, X_d$ containing n^2 variables each and these sets are viewed as $n \times n$ matrices. The polynomial $\text{IMM}_{n,d}$ is defined as the $(1, 1)$-th entry of the matrix product $X_1 \cdots X_d$.

Definition 1.3: ABP

An **ABP** is a directed layered graph with edges from one layer to the next layer. Every edge is labelled with a weight which is a linear polynomial $(c_0 + \sum_{i=1}^{n} c_i x_i)$ for $c_i \in \mathbb{F}$. The first layer has a single vertex s called the source and the last layer has a single vertex t called the sink. The polynomial computed by the ABP is

$$\sum_{P \text{ is a path from s to t}} \text{weight}(P)$$

where $\text{weight}(P)$ denotes the product of the edge-weights lying on the path P.

1.2 VP and VNP: Algebraic Complexity Classes

We need algebraic complexity classes to classify polynomials based on their computational complexity in terms of these algebraic models of computation. Valiant [Val79], in a very influential work defined the classes VP and VNP which can be considered the arithmetic analogues of P and NP.

Definition 1.4: VP

A family of polynomials (f_n) is said to be in the class **VP** if each f_n is a $p(n)$-variate polynomial of degree $q(n)$ for some polynomially bounded functions p and q and it is computable by a circuit of size polynomially bounded in n.

Definition 1.5: VNP

A family of polynomials (f_n) is said to be in the class **VNP** if there exist polynomially bounded functions p and q and a family of polynomials (g_n) $\in$VP of polynomials $g_n \in \mathbb{F}[x_1, \cdots, x_{p(n)}, y_1, \cdots, y_{q(n)}]$ such that

$$f_n(x_1, \cdots, x_{p(n)}) = \sum_{e \in \{0,1\}^{q(n)}} g_n(x_1, \cdots, x_{p(n)}, e_1, \cdots, e_{q(n)}).$$

Clearly, VP $\subseteq$ VNP. Much like the P vs NP problem in the Boolean world, the central open problem of algebraic complexity theory is to separate VP from VNP i.e. to exhibit a polynomial family in VNP which requires superpolynomial sized general algebraic circuits to be computed.

But there are some interesting *depth reduction* results which show that depth 3 and depth 4 circuits are almost as powerful as general ones.

Lemma 1.1: Depth reduction [VSBR83, AV08, Koi12, Tav13, GKKS16]

Let f be an n-variate degree d polynomial computed by a size s arithmetic circuit. Then f can be computed by a depth four circuit of size $s^{O(\sqrt{d})}$. If this polynomial f is over $\mathbb{Q}$, then it can also be computed by a depth three circuit of size $s^{O(\sqrt{d})}$.

Hence proving an $n^{\omega(\sqrt{d})}$ lower bound on these special circuits is enough to separate VP from VNP. This is our motivation to study constant depth circuit lower bounds.

1.3 Before 2021: Lower Bounds for Constant Depth Circuits

In the Boolean world, strong lower bound for constant depth circuits were known since the 1980's [FSS81, Ajt83, Has86, Raz87, Smo87], but for constant depth algebraic circuits, superpolynomial lower bounds remained elusive for a long time. Till 2021, the best known lower bound for even depth 3 circuits was near cubic. [KST16] proved a lower bound of $\Omega(n^3/(\log n)^2)$ against depth 3 circuits. In [GST20], a lower bound of $\Omega(n^{2.5}/(\log n)^6)$ was obtained for depth 4 circuits. For a general constant Δ, a lower bound of the form $n^{1+\Omega(1/\Delta)}$ was known for algebraic circuits of depth Δ [SS97, Raz10]. Clearly, these lower bounds fall far short of the superpolynomial lower bounds we hope to prove.

1.4 2021: The LST Breakthrough

In 2021, Limaye, Srinivasan and Tavenas [LST] proved the first-ever superpolynomial lower bound for general constant-depth circuits. More precisely, they showed that the Iterated Matrix Multiplication polynomial $\mathrm{IMM}_{n,d}$ (where $d = o(\log n)$) has no product-depth Δ circuits of size $n^{d^{\exp(-O(\Delta))}}$. Note that for any $\Delta \leq \log d$, $\mathrm{IMM}_{n,d}$ has a set-multilinear circuit of product-depth Δ and size $n^{O(d^{1/\Delta})}$, obtained by simple divide-and-conquer approach.

The lower bound proof of [LST] proceeds in two steps:

- **Set-multilinearization:** In the first step, we show that if a set-multilinear polynomial has a circuit of depth Δ and size s, then it can also be computed by a set-multilinear circuit of depth at most 2Δ and size $d^{O(d)}poly(s)$. As the blowup in size only depends on d, we can work in the low-degree regime (take $d = O(\log n / \log\log n)$) and here a superpolynomial lower bound for constant-depth set-multilinear circuits implies a superpolynomial lower bound for general constant-depth circuit.
- **Set-multilinear lower bound:** In this step, we prove a lower bound of the form $n^{d^{\exp(-O(\Delta))}}$ for set-multilinear circuits of constant depth Δ, using the so-called *partial derivative method*, used first in [NW95] to obtain set-multilinear circuit lower bounds. This method was applied in [LST] with the important change that the sets $X_1, \cdots, X_d$ were now allowed to be of *different sizes* and this discrepancy in set sizes crucially helps in getting strong set-multilinear lower bounds.

1.5 Some More Recent Works

In a further recent work [TLS], Tavenas, Limaye and Srinivasan proved a product-depth Δ *set-multilinear* formula lower bound of $(\log n)^{\Omega(\Delta d^{1/\Delta})}$ for $\mathrm{IMM}_{n,d}$. There is no restriction of degree, but in the small degree regime, the bound is much weaker than [LST] and cannot be used for escalation. Improving on it, Kush and Saraf [KS] showed a lower bound of $n^{\Omega(n^{1/\Delta}/\Delta)}$ for the size of product-depth Δ set-multilinear formulas computing an n^2-variate, degree n polynomial in VNP from the family of

Nisan-Wigderson design-based polynomials. Kush and Saraf further improved the result in [KS23] by proving the same lower bound for a $\Theta(n^2)$-variate, degree $\Theta(n)$ polynomial which is computable by a set-multilinear ABP of polynomial size.

1.6 Contribution of this Thesis

In this thesis, we see an improved lower bound for IMM against general constant depth circuits.

For the rest of this paper, let $F(n) = \Theta(\varphi^n)$ be the n-th Fibonacci number (starting with $F(0) = 1$, $F(1) = 2$) where $\varphi = (1 + \sqrt{5})/2 = 1.618\ldots$ is the golden ratio. We define the functions G and μ as $G(n) = F(n) - 1$ and $\mu(n) = 1/G(n) = 1/(F(n) - 1)$ for non-negative integers n.

Theorem 1.1: General circuit lower bound

Fix a field $\mathbb{F}$ of characteristic 0 or characteristic $> d$. Let N, d, Δ be such that $d = o(\log N / \log \log N)$. Then, any product-depth Δ circuit computing $\mathrm{IMM}_{n,d}$ on $N = dn^2$ variables must have size at least $N^{\Omega\left(d^{\mu(2\Delta)}/\Delta\right)}$.

Theorem 1.1 improves on the lower bound of $N^{\Omega\left(d^{1/(2^{2\Delta}-1)}/\Delta\right)}$ of [LST] since $F(2\Delta) = \Theta(\varphi^{2\Delta}) \ll 2^{2\Delta}$.

To prove Theorem 1.1, we use the hardness escalation given by Lemma 2.2 which allows for conversion of general circuits to set-multilinear ones without significant blow up in size (provided the degree is small). The actual lower bound is for set-multilinear circuits.

Theorem 1.2: Set-multilinear circuit lower bound

Let $d \leq (\log n)/4$. Any product-depth Δ *set-multilinear* circuit computing $\mathrm{IMM}_{n,d}$ must have size at least $n^{\Omega\left(d^{\mu(\Delta)}/\Delta\right)}$.

This is an improvement over the $n^{\Omega\left(d^{1/(2^{\Delta}-1)}/\Delta\right)}$ bound of [LST, Lemma 15]. Moreover, the result holds over any field $\mathbb{F}$. The restriction on the characteristic in Theorem 1.1 comes from the conversion to set-multilinear circuits. The difference between $\mu(2\Delta)$ in Theorem 1.1 and $\mu(\Delta)$ in Theorem 1.2 is also due to the doubling of product-depth during this conversion.

Chapter 2

Preliminaries

For any positive integer n, we denote by $F(n)$ the n-th Fibonacci number with $F(0) = 1$, $F(1) = 2$ and $F(n) = F(n-1) + F(n-2)$. The function $G : \mathbb{N} \to \mathbb{N}$ is given by $G(n) = F(n) - 1$. The nearest integer to any real number r is denoted by $\lfloor r \rceil$. We follow the notation of [LST] as much as possible for better readability.

2.1 Words

Words are basically tuples $(w_1, \ldots, w_d)$ of length d where $2^{|w_i|}$ are integers. These words define the actual set sizes of the set-multilinear polynomials we will be working with. Given a word w, let $\overline{X}(w)$ denote the tuple of sets of variables $(X_1(w), \ldots, X_d(w))$ where the size of each $X_i(w)$ is $2^{|w_i|}$. We denote the space of set-multilinear polynomials over $\overline{X}(w)$ by $\mathbb{F}_{sm}[\overline{X}(w)]$.

For a word w and any subset $S \subseteq [d]$, the sum of elements of w indexed by S is denoted by $w_S = \sum_{i \in S} w_i$. For all $t \leq d$, if it holds that $|w_{[t]}| \leq b$, then we call w 'b-unbiased'. Denote by $w_{|S}$ the sub-word indexed by S. The positive and negative indices of w are denoted $\mathcal{P}_w = \{i \mid w_i \geq 0\}$ and $\mathcal{N}_w = \{i \mid w_i < 0\}$ respectively with the corresponding collections $\{X_i(w)\}_{i \in \mathcal{P}_w}$ and $\{X_i(w)\}_{i \in \mathcal{N}_w}$ being the positive and negative variable sets. We denote by $\mathcal{M}_w^{\mathcal{P}}$ (resp. $\mathcal{M}_w^{\mathcal{N}}$) the set of all set-multilinear monomials over the positive (resp. negative) variable sets.

2.2 Relative Rank: The Complexity Measure

The *partial derivative matrix* $\mathcal{M}_w(f)$ of $f \in \mathbb{F}_{sm}[\overline{X}(w)]$ has rows indexed by $\mathcal{M}_w^{\mathcal{P}}$ and columns by $\mathcal{M}_w^{\mathcal{N}}$. The entry corresponding to row $m_+ \in \mathcal{M}_w^{\mathcal{P}}$ and $m_- \in \mathcal{M}_w^{\mathcal{N}}$ is the coefficient of the monomial m_+m_- in f. The complexity measure we use is the *relative rank*, same as [LST]:

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

The following properties of relrk_w will be useful.

1. (Imbalance) For any $f \in \mathbb{F}_{sm}[\overline{X}(w)]$, $\mathrm{relrk}_w(f) \leq 2^{-|w_{[d]}|/2}$.
2. (Sub-additivity) For any $f, g \in \mathbb{F}_{sm}[\overline{X}(w)]$, $\mathrm{relrk}_w(f+g) \leq \mathrm{relrk}_w(f) + \mathrm{relrk}_w(g)$.
3. (Multiplicativity) Suppose $f = f_1 f_2 \cdots f_t$ where $f_i \in \mathbb{F}_{sm}[\overline{X}(w_{|S_i})]$ and $(S_1, \ldots, S_t)$ is a partition of $[d]$. Then, $\mathrm{relrk}_w(f) = \mathrm{relrk}_w(f_1 f_2 \cdots f_t) = \prod_{i\in[t]} \mathrm{relrk}_{w_{|S_i}}(f_i)$.

For sake of completion, we provide the proof from [LST].

Proof. 1. We have $|\mathcal{M}_w^{\mathcal{P}}| = 2^{\sum_{i\in\mathcal{P}_w} w_i}$ and $|\mathcal{M}_w^{\mathcal{N}}| = 2^{-\sum_{i\in\mathcal{N}_w} w_i}$. Hence,

$$\mathrm{relrk}_w(f) \leq \frac{\min\left(|\mathcal{M}_w^{\mathcal{P}}|, |\mathcal{M}_w^{\mathcal{N}}|\right)}{2^{\frac{1}{2}\sum_{i\in[d]}|w_i|}} = \sqrt{\frac{\min\left(|\mathcal{M}_w^{\mathcal{P}}|, |\mathcal{M}_w^{\mathcal{N}}|\right)}{\max\left(|\mathcal{M}_w^{\mathcal{P}}|, |\mathcal{M}_w^{\mathcal{N}}|\right)}} = 2^{-|w_{[d]}|/2} \ .$$

2. $\mathcal{M}_w(f+g) = \mathcal{M}_w(f) + \mathcal{M}_w(g) \implies \mathrm{rank}(\mathcal{M}_w(f+g)) \leq \mathrm{rank}(\mathcal{M}_w(f)) + \mathrm{rank}(\mathcal{M}_w(g))$, which implies the subadditivity property of relative rank.

3. The matrix $\mathcal{M}_w(f)$ equals to the Kronecker product $\mathcal{M}_w(f_1) \otimes \cdots \otimes \mathcal{M}_w(f_t)$. Therefore,

$$\text{relrk}_w(f) = \frac{\prod_{i\in[t]} \text{rank}(\mathcal{M}_w(f_i))}{\prod_{i\in[t]} 2^{\frac{1}{2}\sum_{j\in S_i}|w_j|}} = \prod_{i\in[t]} \text{relrk}_{w_{|S_i}}(f_i) \ .$$

□

2.3 Word Polynomials

We now define the hard polynomials we prove lower bounds for. For any monomial $m \in \mathbb{F}_{sm}[\overline{X}(w)]$, let $m_+ \in \mathcal{M}_w^{\mathcal{P}}$ and $m_- \in \mathcal{M}_w^{\mathcal{N}}$ be its "positive" and "negative" parts. As $|X_i| = 2^{|w_i|}$, the variables of X_i can be indexed using boolean strings of length $|w_i|$. This gives a way to associate a boolean string with any monomial. Let $\sigma(m_+)$ and $\sigma(m_-)$ be the strings associated with m_+ and m_- respectively. We write $\sigma(m_+) \sim \sigma(m_-)$ if one is a prefix of the other.

Definition 2.1: Word Polynomials [LST]

Let w be any word. The polynomial P_w is defined as the sum of all monomials $m \in \mathbb{F}_{sm}[\overline{X}(w)]$ such that $\sigma(m_+) \sim \sigma(m_-)$.

The matrices $M_w(P_w)$ have full rank (equal to either the number of rows or columns, whichever is smaller) and hence $\text{relrk}_w(P_w) = 2^{-|w_{[d]}|/2}$. We note (without proof) that these polynomials can be obtained as *set-multilinear* restrictions of $\text{IMM}_{n,d}$.

Lemma 2.1: [LST, Lemma 8]

Let w be any b-unbiased word. If there is a set-multilinear circuit computing $\mathrm{IMM}_{2^b,d}$ of size s and product-depth Δ, then there is also a set-multilinear circuit of size s and product-depth Δ computing the polynomial $P_w \in \mathbb{F}_{sm}[\overline{X}(w)]$. Moreover, $\mathrm{relrk}_w(P_w) \geq 2^{-b/2}$.

The following lemma from [LST] tells us that any circuit over a large characteristic field can be *set-multilinearized* with a blowup in depth by a factor of 2 and a blowup in size by a factor which is exponential only in $poly(d)$.

Lemma 2.2: [LST, Proposition 9]

Let s, N, d, Δ be growing parameters with $s \geq Nd$. Assume that $char(\mathbb{F}) = 0$ or $char(\mathbb{F}) > d$. If C is a circuit of size at most s and product-depth at most Δ computing a set-multilinear polynomial P over the sets of variables $(X_1, \ldots, X_d)$ (with $|X_i| \leq N$), then there is a *set-multilinear circuit* $\tilde{C}$ of size $d^{O(d)}\mathrm{poly}(s)$ and product-depth at most 2Δ computing the same polynomial P.

Hence, we can restrict ourselves to work in the low-degree regime so that the blowup in size is at most polynomial in s.

Chapter 3

Lower bound proof overview

In this chapter, we provide a proof overview of Theorem 1.2 for depth three circuits. Then we discuss the obstacles in extending this proof strategy to higher-depth circuits and the ideas used in overcoming these obstacles.

By Lemma 2.2, our goal is to prove set-multilinear circuit lower bounds for the IMM polynomial. Lemma 2.1 says that it suffices to prove the set-multilinear circuit lower bound for a word polynomial P_w. This lemma also tells us that if a word w is k-unbiased for some small k, then the polynomial P_w has high relative rank. Therefore, if we can choose such a word w and show that for this choice of word (and hence set sizes), the relative rank is small for set-multilinear circuits of a certain size, we will be done.

Let k be an integer close to $\log_2 n$.

Word chosen in [LST]: The positive entries of the word w were equal to an integer close to $k/\sqrt{2}$ and the negative entries were $-k$. Evidently, these entries are independent of the product-depth Δ.

Word chosen in this thesis: The positive entries of the word w are $(1 - p/q)k$ and the negative entries are $-k$ where p and q are suitable integers dependent on

Δ. This *depth-dependent* construction of the word enables us to improve the lower bound.

We demonstrate the high level proof strategy of the lower bound for the case of product-depth 3.

3.1 Proof overview of Theorem 1.2 for $\Delta = 3$

Define $\lambda = \lfloor d^{1/G(3)} \rfloor$. Consider a set-multilinear formula C of product-depth 3 and let v be a gate in it. Suppose that the subformula $C^{(v)}$ rooted at v has product-depth $\delta \leq 3$, size s and degree $\geq \lambda^{G(\delta)}/2$. We will prove that $\text{relrk}_w(C^{(v)}) \leq s2^{-k\lambda/48}$ by induction on δ. This will give us the desired upper bound of the form $s2^{-k\lambda/48} = sn^{-\Omega(d^{\mu(3)})}$ on the relative rank of the whole formula when v is taken to be the output gate.

Write $C^{(v)} = C_1 + \cdots + C_t$ where each C_i is a subformula of size s_i rooted at a product gate. Because of the subadditivity of relrk_w, it suffices to show that $\text{relrk}_w(C_i) \leq s_i 2^{-k\lambda/48}$ for all i.

Base case: If $\delta = 1$, then C_i is a product of linear forms. Thus, it has rank 1 and hence low relative rank.

Induction step: $\delta \in \{2, 3\}$. Write $C_i = C_{i,1} \ldots C_{i,t_i}$ where each $C_{i,j}$ is a subformula of product-depth $\delta - 1$. If any $C_{i,j}$ has degree $\geq \lambda^{G(\delta-1)}/2$, then by induction hypothesis, the relative rank of $C_{i,j}$ and hence C_i will have the desired upper bound and we are done.

Otherwise each $C_{i,j}$ has degree $D_{ij} < \lambda^{G(\delta-1)}/2$. As the formula is set-multilinear, there is a collection of variable-sets $(X_l)_{l \in S_j}$ with respect to which $C_{i,j}$ is set-multilinear. For $j \in [t_i]$, let a_{ij} be the number of positive indices in S_j i.e. the number of positive sets in the collection $(X_l)_{l \in S_j}$. Then the number of negative indices is $(D_{ij} - a_{ij})$.

We consider two cases: if $a_{ij} \leq D_{ij}/3$, then $w_{S_j} \leq (D_{ij}/3)\cdot(1-p/q)k+(2D_{ij}/3)\cdot(-k)$ $\leq -D_{ij}k/3$. Otherwise $a_{ij} > D_{ij}/3$ and if we can prove that $|w_{S_j}| \geq a_{ij}k/(4\lambda^{G(\delta)-1})$, then in both of the above cases, we would have $|w_{S_j}| \geq D_{ij}k/(12\lambda^{G(\delta)-1})$. By the multiplicativity and imbalance property of relrk_w, it would follow that $\mathrm{relrk}_w(C_i) \leq 2^{\sum_{j=1}^{t_i} -\frac{1}{2}|w_{S_j}|} \leq 2^{-k\lambda/48}$ and we would be done. Thus, we now only have to show that $|w_{S_j}| \geq a_{ij}k/(4\lambda^{G(\delta)-1})$. We have

$$|w_{S_j}| = |a_{ij}(1-p/q) - (D_{ij} - a_{ij})|\, k\ .$$

Notice that $|w_{S_j}|/k$ is the distance of $a_{ij}p/q$ from some integer, so it must be at least the minimum of $\{a_{ij}p/q\}$ and $1 - \{a_{ij}p/q\}$ where $\{.\}$ denotes the fractional part. The number $a_{ij}p/q$ being rational, has a fractional part $\zeta = (a_{ij}p \bmod q)/q$ and hence it comes down to finding a nice tuple (p, q) which satisfies the following system of inequalities:

$$\min(\zeta,\ 1-\zeta) \geq a_{ij}/(4\lambda^{G(\delta)-1}) \text{ for } \delta \in \{2,3\} \text{ when } a_{ij} \leq D_{ij} < \lambda^{G(\delta-1)}/2\ .$$

This notion is captured by the definition of (d, Δ)-niceness of a tuple (p, q) in Chapter 4.

Here, assign $p = \lambda$, $q = \lambda^2 + 1$.

The inequality for the $\delta = 2$ case is clearly satisfied as $(a_{ij}\lambda \mod (\lambda^2+1)) = a_{ij}\lambda$ when $0 \leq a_{ij} \leq \lambda/2$.

Consider the case of $\delta = 3$ and $a_{ij} < \lambda^2/2$. Write $a_{ij} = y_1\lambda + y_0$ for integers $y_1 = \lfloor a_{ij}/\lambda \rfloor < \lambda/2$ and $y_0 \leq \lambda - 1$. Thus, $a_{ij}\lambda \equiv -y_1 + y_0\lambda \mod (\lambda^2+1)$. Through some case analysis, one can show that $\min\Big(|y_0\lambda - y_1|,\ \lambda^2+1-|y_0\lambda-y_1|\Big) \geq y_1$ which immediately implies the inequality for the $\delta = 3$ case as $y_1 = \lfloor a_{ij}/\lambda \rfloor \geq a_{ij}/(2\lambda)$.

3.2 Obstacles in extending the above proof strategy to product-depth 4 and how to overcome them

Obstacle: We can attempt to extend the above proof technique to product-depth 4 as follows:

We would similarly want to express a_{ij} as $a_{ij} = y_2\lambda^2 + y_1\lambda + y_0$ for integers $y_2 = \lfloor a_{ij}/\lambda^2 \rfloor, y_0 \leq \lambda - 1$ and $y_1 \leq \lambda - 1$. Ideally, we would want that for some $q \approx \lambda^4$,

$$p\lambda^2 \equiv 1 \bmod q,\ p\lambda \equiv -\lambda^2 \bmod q \text{ and } p \equiv \lambda^3 \bmod q$$

so that $a_{ij}p \equiv y_2 - y_1\lambda^2 + y_0\lambda^3 \bmod q$ and then we can carry out a similar analysis as in the $\Delta = 3$ case. But this is not possible since multiplying the second congruence equation by λ gives $p\lambda^2 \equiv -\lambda^3 \bmod q$, which contradicts the first congruence equation.

Workaround: We decide to express a_{ij} as $a_{ij} = y_2b_2 + y_1b_1 + y_0b_0$ where b_2, b_1, b_0 are close to $\lambda^2, \lambda, 1$ respectively, instead of being precisely equal to these powers of λ. Then we choose $c_2 \approx 1, c_1 \approx -\lambda^2, c_0 \approx \lambda^3$ and we assign values to p and q such that

$$pb_2 \equiv c_2 \bmod q,\ pb_1 \equiv c_1 \bmod q \text{ and } pb_0 \equiv c_0 \bmod q.$$

It is easy to verify that all these conditions are satisfied if we define
$b_0 = 1, b_1 = \lambda, b_2 = b_1(\lambda - 1) + b_0;$ $\qquad c_2 = 1, c_1 = -\lambda^2, c_0 = c_2 - c_1(\lambda - 1);$
$p = c_0$ and $q = pb_1 - c_1$.
This inspired our construction of the sequences $\{b_m\}$ and $\{c_m\}$ for general product-depth Δ.

Chapter 4

Improved lower bound for constant depth circuits

In this chapter, we prove Theorem 1.1 and Theorem 1.2.

Theorem 1.1: General circuit lower bound

Fix a field $\mathbb{F}$ of characteristic 0 or characteristic $> d$. Let N, d, Δ be such that $d = o(\log N / \log \log N)$. Then, any product-depth Δ circuit computing $\mathrm{IMM}_{n,d}$ on $N = dn^2$ variables must have size at least $N^{\Omega\left(d^{\mu(2\Delta)}/\Delta\right)}$.

Theorem 1.2: Set-multilinear circuit lower bound

Let $d \leq (\log n)/4$. Any product-depth Δ *set-multilinear* circuit computing $\mathrm{IMM}_{n,d}$ must have size at least $n^{\Omega\left(d^{\mu(\Delta)}/\Delta\right)}$.

4.1 Proof of the Lower Bounds

We first prove Theorem 1.1 in the same style as the proof of [LST, Corollary 4]:

Proof of Theorem 1.1. From Lemma 2.2 and Theorem 1.2, for a circuit of product-depth Δ and size s computing $\text{IMM}_{n,d}$, we get that

$$d^{O(d)}\text{poly}(s) \geq N^{\Omega\left(d^{\mu(2\Delta)}/2\Delta\right)}.$$

Since $d = O(\log N/\log\log N)$, it follows that $d^{O(d)} = N^{O(1)}$. Therefore,

$$\text{poly}(s) \geq N^{\Omega\left(d^{\mu(2\Delta)}/2\Delta\right)}/d^{O(d)} \geq N^{\Omega\left(d^{\mu(2\Delta)}/4\Delta\right)}$$

implying the required lower bound on s and thus, Theorem 1.1. □

Now we prove Theorem 1.2. To do this, we need the notion of $(d,\Delta)-$nice tuples of integers, defined as follows.

Definition 4.1: $(d,\Delta)-$niceness

Let d,Δ be positive integers and let $\lambda := \lfloor d^{1/G(\Delta)} \rfloor$. Then, a tuple of positive integers (p,q) is called $(d,\Delta)-$nice if it satisfies the following two conditions:

- **Condition 1:** $q \leq d$ and $\dfrac{1}{2\lambda} \leq \dfrac{p}{q} \leq \dfrac{1}{2}$.
- **Condition 2:** for all $\delta \in \{2,\cdots,\Delta\}$, for all positive integers $z < \lambda^{G(\delta-1)}/8$,

$$\min\left(\frac{zp \bmod q}{q}, 1 - \frac{zp \bmod q}{q}\right) \geq \frac{z}{8\lambda^{G(\delta)-1}}.$$

Basically, if we have such a tuple (p,q), then we can define the variable set sizes in terms of this tuple and the above-mentioned properties of this tuple will ensure that the discrepancy in the set sizes is *nice enough* to obtain strong set-multilinear lower bounds. The following lemma guarantees the existence of such tuples in most cases:

Lemma 4.1: Existence of (d, Δ)-nice tuples

For every pair of positive integers d, Δ satisfying $\lfloor d^{1/G(\Delta)} \rfloor \geq 3$, there exists a tuple of positive integers (p, q) which is (d, Δ)-nice.

We devote Section 4.2 to the proof of this lemma.

Proof of Theorem 1.2. Fix the product-depth Δ for which we want to prove the set-multilinear formula lower bound. Define $\lambda := \lfloor d^{1/G(\Delta)} \rfloor$. If $\lambda \geq 3$, then $d^{\mu(\Delta)} < 3$ and in that case, the lower bound is trivial. Hence, we can assume that $\lambda \geq 3$. By Lemma 4.1, there exists a tuple of positive integers (p, q) which is (d, Δ)-nice. Using these numbers p, q, we first construct a word w' such that the word polynomial $P_{w'}$ is hard to compute.

Construction of the word: Define $\alpha = 1 - p/q$.

By the first condition of (d, Δ)-niceness for the tuple (p, q), we know that $\alpha \geq 1/2$ and

$$q \leq d < \lfloor \log_2 n \rfloor / 2\ .$$

Therefore, there exists a multiple of q in the interval $\left[\frac{\lfloor \log_2 n \rfloor}{2}, \lfloor \log_2 n \rfloor\right]$. Let k be this multiple of q.

Then αk is an integer. We can construct a word w' over the alphabet $\{\alpha k, -k\}$ such that w' is k-unbiased. This can be done using induction: set ${w'}_1 := -k$. At the i-th step, if $|{w'}_{[i]}| \leq 0$, set ${w'}_{i+1} := \alpha k$, otherwise set ${w'}_{i+1} := -k$.

Assume the following lemma:

Lemma 4.2

Let $\delta \leq \Delta$ be an integer and α, k be as defined above. Let w be any word of length d over the alphabet $\{\alpha k, -k\}$. Then any set-multilinear formula C of product-depth δ, degree $D \geq \lambda^{G(\delta)}/8$ and size at most s satisfies

$$\text{relrk}_w(C) \leq s2^{-k\lambda/256}.$$

By Lemma 2.1, there exists a set-multilinear projection $P_{w'}$ of $\text{IMM}_{2^k,d}$ such that $\text{relrk}_{w'}(P_{w'}) \geq 2^{-k}$. If there is a set-multilinear circuit of size s and product-depth Δ computing $\text{IMM}_{n,d}$, then we can expand it to a set-multilinear formula of size at most $s^{2\Delta}$ which computes the same polynomial. Hence we will also have a set-multilinear formula of size at most $s^{2\Delta}$ computing $P_{w'}$. As $d \geq \lambda^{G(\Delta)}/8$, taking the particular case of $\delta = \Delta$ in Lemma 4.2, we obtain $\text{relrk}_{w'}(P_{w'}) \leq s^{2\Delta}2^{-k\lambda/256}$. This gives the desired lower bound

$$s^{2\Delta} \geq 2^{-k}2^{k\lambda/256} \geq \left(\frac{n}{4}\right)^{\frac{d^{1/G(\Delta)}}{512}} /n = n^{\Omega(d^{\mu(\Delta)})}.$$

□

Proof of Lemma 4.2. We proceed by induction on δ. We can write $C = C_1 + \cdots + C_t$ where each C_i is a subformula of size s_i rooted at a product gate. Because of the subadditivity of relrk_w, it suffices to show that

$$\text{relrk}_w(C_i) \leq s_i 2^{-k\lambda/256} \quad \text{for all } i.$$

Base case: C has product-depth $\delta = 1$ and degree $D \geq \lambda/8$.

Then C_i is a product of linear forms. If L is linear form on some variable set $X(w_j)$, then $\text{relrk}_w(L) \leq 2^{-|w_j|/2} \leq 2^{-k/4}$. Therefore by the multiplicativity of relrk_w,

$$\text{relrk}_w(C_i) \leq 2^{-kD/4} \leq 2^{-k\lambda/32} .$$

Induction hypothesis: Assume that the lemma is true for all product-depths $\leq \delta - 1$.

Induction step: Let C be a formula of product-depth δ and degree $D \geq \lambda^{G(\delta)}/8$.

We can write $C_i = C_{i,1} \dots C_{i,t_i}$ where each $C_{i,j}$ is a subformula of product-depth $\delta - 1$.

If C_i has a factor, say $C_{i,1}$, of degree $\geq \lambda^{G(\delta-1)}/8$, then by induction hypothesis,

$$\text{relrk}_w(C_i) \leq \text{relrk}_w(C_{i,1}) \leq s_i 2^{-k\lambda/256} \ .$$

Otherwise every factor of C_i has degree $< \lambda^{G(\delta-1)}/8$. Let $C_i = C_{i,1} \dots C_{i,t_i}$ where each $C_{i,j}$ has degree $D_{ij} < \lambda^{G(\delta-1)}/8$. If C_i is set-multilinear with respect to $(X_l)_{l \in S}$, then let $(S_1, \dots, S_{t_i})$ be the partition of S such that each $C_{i,j}$ is set-multilinear with respect to $(X_l)_{l \in S_j}$.

For $j \in [t_i]$, let a_{ij} be the number of positive indices in S_j. We have two cases:
Case 1: $a_{ij} \leq D_{ij}/2$

We have

$$\begin{aligned} w_{S_j} &= a_{ij} \cdot \alpha k + (D_{ij} - a_{ij}) \cdot (-k) \\ &\leq \frac{D_{ij}}{2} \cdot \alpha k + \frac{D_{ij}}{2} \cdot (-k) = -\frac{D_{ij} p}{2q} k \leq -\frac{D_{ij} k}{4\lambda} \end{aligned}$$

where the last inequality follows from the first condition of (d, Δ)-niceness for the tuple (p, q). This implies that $|w_{S_j}| \geq \left| \frac{D_{ij} k}{4\lambda} \right| \geq D_{ij} k / (16 \lambda^{G(\delta) - 1})$.

Case 2: $a_{ij} > D_{ij}/2$

We have

$$\begin{aligned} |w_{S_j}| &= |a_{ij} \cdot \alpha k + (D_{ij} - a_{ij}) \cdot (-k)| \\ &= \left| a_{ij} \frac{p}{q} - (2a_{ij} - D_{ij}) \right| k \qquad \text{as } \alpha = 1 - p/q \end{aligned}$$

$$\geq \left| \frac{a_{ij}p}{q} - \left\lfloor \frac{a_{ij}p}{q} \right\rceil \right| k \qquad \text{where } \lfloor . \rceil \text{ denotes the nearest integer.}$$

Now $\left| \frac{a_{ij}p}{q} - \left\lfloor \frac{a_{ij}p}{q} \right\rceil \right|$ can be equal to either the fractional part of $\frac{a_{ij}p}{q}$ or one minus the fractional part. As $\frac{a_{ij}p}{q}$ is a rational number, its fractional part is $\frac{a_{ij}p \bmod q}{q}$. Hence,

$$|w_{S_j}| \geq \min\left(\frac{a_{ij}p \bmod q}{q}, 1 - \frac{a_{ij}p \bmod q}{q}\right) k \ .$$

As $a_{ij} \leq D_{ij} < \lambda^{G(\delta-1)}/8$, it follows from the second condition of (d, Δ)-niceness for the tuple (p, q) that

$$|w_{S_j}| \geq \frac{a_{ij}k}{8\lambda^{G(\delta)-1}} > \frac{D_{ij}k}{16\lambda^{G(\delta)-1}} \ .$$

Hence in both of the above cases, we have $|w_{S_j}| \geq D_{ij}k/(16\lambda^{G(\delta)-1})$. By the multiplicativity and imbalance property of relrk_w and the assumption $D \geq \lambda^{G(\delta)}/8$, it follows that

$$\text{relrk}_w(C_i) \leq \prod_{j=1}^{t_i} 2^{-\frac{1}{2}|w_{S_j}|} \leq 2^{-\sum_{j=1}^{t_i} D_{ij}k/(32\lambda^{G(\delta)-1})} = 2^{-Dk/(32\lambda^{G(\delta)-1})} \leq 2^{-k\lambda/256} \ .$$

□

4.2 Existence of (d, Δ)-nice tuples

In this section, we prove Lemma 4.1.

For the rest of the section, let $\lambda = \lfloor d^{1/G(\Delta)} \rfloor \geq 3$. We will construct two sequences $\{b_m\}$ and $\{c_m\}$ of integers which satisfy some nice properties. Then we will use these sequences to define our (d, Δ)-nice tuple (p, q). The nice properties of these sequences will help us in proving the (d, Δ)-niceness of (p, q).

4.2.1 Defining the sequences $\{b_m\}$, $\{c_m\}$ and the tuple (p, q):

Let $r_m := \lambda^{G(m+1)-G(m)} - 1$ for $0 \leq m \leq \Delta - 2$.

Define

$$b_0 := 1, \quad b_1 := \lambda \text{ and } b_m := b_{m-2} + r_{m-1}b_{m-1} \text{ for } 2 \leq m \leq \Delta - 2\ .$$

Define

$$c_{\Delta-2} := (-1)^{\Delta-2}, \quad c_{\Delta-3} := (-1)^{\Delta-3}\lambda^{G(\Delta-1)-G(\Delta-2)} \text{ and}$$
$$c_m := (-1)^m(|c_{m+2}| + r_{m+1}|c_{m+1}|) \text{ for } \Delta - 4 \geq m \geq 0\ .$$

Note that the sign parity of c_m is $(-1)^m$ i.e. $|c_m| = (-1)^m c_m$ for all m.

Thus,

$$\begin{aligned} c_{m-2} &= (-1)^{m-2}(|c_m| + r_{m-1}|c_{m-1}|) \\ &= (-1)^{m-2}((-1)^m c_m + r_{m-1} \cdot (-1)^{m-1} c_{m-1}) \\ &= c_m - r_{m-1}c_{m-1} \end{aligned}$$

which implies

$$c_m = c_{m-2} + r_{m-1}c_{m-1} \text{ for } 2 \leq m \leq \Delta - 2\ .$$

Define

$$p := c_0 \text{ and } q := pb_1 - c_1 = c_0(r_0 + 1) - c_1\ .$$

By defining the integers p and q this way, we have ensured that $pb_0 \equiv c_0 \bmod q$ and $pb_1 \equiv c_1 \bmod q$. Hence from the relations $b_m = b_{m-2} + r_{m-1}b_{m-1}$ and $c_m = c_{m-2} + r_{m-1}c_{m-1}$, it inductively follows that

$$pb_m \equiv c_m \bmod q \qquad \text{for } 0 \leq m \leq \Delta - 2\ . \tag{4.1}$$

4.2.2 Bounds on the values of b_m and $|c_m|$

To prove the bounds, we need a generalized version of the well-known Bernoulli's inequality [Mit70, Section 2.4]:

Claim 4.1 (Bernoulli's inequality)**.** *Let $x_1, \ldots, x_r$ be real numbers all greater than -1 and all with the same sign. Then,*

$$(1 + x_1)(1 + x_2) \ldots (1 + x_r) \geq 1 + x_1 + \ldots + x_r\,.$$

Proof. We prove it by induction on r. The base case $r = 1$ is trivial.

Assume that $(1 + x_1)(1 + x_2) \ldots (1 + x_{r-1}) \geq 1 + x_1 + \ldots + x_{r-1}$. Then,

$$\begin{aligned}(1 + x_1)(1 + x_2) \ldots (1 + x_r) &\geq (1 + x_1 + \ldots + x_{r-1})(1 + x_r)\\ &= (1 + x_1 + \ldots + x_r) + (x_1x_r + x_2x_r + \ldots + x_{r-1}x_r)\\ &\geq 1 + x_1 + \ldots + x_r\end{aligned}$$

where the last inequality follows from the fact that all the x_i's are of the same sign. □

Each b_m is close to $\lambda^{G(m)}$ and each $|c_m|$ is close to $\lambda^{G(\Delta-1)-G(m+1)}$:

Lemma 4.3

For $0 \leq m \leq \Delta - 2$,we have $\dfrac{\lambda^{G(m)}}{2} \leq b_m \leq \lambda^{G(m)}$ and $\dfrac{\lambda^{G(\Delta-1)-G(m+1)}}{2} \leq |c_m| \leq \lambda^{G(\Delta-1)-G(m+1)}$.

Proof. Clearly, b_m satisfies the bounds when $m = 0$ or 1. For $m \geq 2$,

$$\begin{aligned}
b_m &= (\lambda^{G(m)-G(m-1)} - 1)b_{m-1} + b_{m-2} \\
&\leq \lambda^{G(m)-G(m-1)} b_{m-1} \\
&\leq \lambda^{G(m)-G(m-1)} . \lambda^{G(m-1)-G(m-2)} \dots \lambda^{G(2)-G(1)} b_1 \\
&= \lambda^{G(m)}.
\end{aligned}$$

$$\begin{aligned}
b_m &= (\lambda^{G(m)-G(m-1)} - 1)b_{m-1} + b_{m-2} \\
&\geq (\lambda^{G(m)-G(m-1)} - 1)b_{m-1} \\
&\geq (\lambda^{G(m)-G(m-1)} - 1).(\lambda^{G(m-1)-G(m-2)} - 1) \dots (\lambda^{G(2)-G(1)} - 1)b_1 \\
&= \lambda^{G(m)-G(1)} b_1 . \left(1 - \frac{1}{\lambda^{G(m)-G(m-1)}}\right)\left(1 - \frac{1}{\lambda^{G(m-1)-G(m-2)}}\right) \dots \left(1 - \frac{1}{\lambda^{G(2)-G(1)}}\right) \\
&\geq \lambda^{G(m)} . \left(1 - \frac{1}{\lambda^{G(m)-G(m-1)}} - \frac{1}{\lambda^{G(m-1)-G(m-2)}} - \dots - \frac{1}{\lambda^{G(2)-G(1)}}\right) \text{ [By Claim 4.1]} \\
&\geq \lambda^{G(m)} . \left(1 - \frac{1}{\lambda^{m-1}} - \frac{1}{\lambda^{m-2}} - \dots - \frac{1}{\lambda}\right) \\
&= \lambda^{G(m)} . \left(1 - \frac{1}{\lambda - 1}\left(1 - \frac{1}{\lambda^{m-1}}\right)\right) \geq \frac{\lambda^{G(m)}}{2}.
\end{aligned}$$

Clearly, $|c_m|$ satisfies the bounds when $m = \Delta - 2$ or $\Delta - 3$. For $m \leq \Delta - 4$,

$$\begin{aligned}
|c_m| &= (\lambda^{G(m+2)-G(m+1)} - 1)|c_{m+1}| + |c_{m+2}| \\
&\leq \lambda^{G(m+2)-G(m+1)} |c_{m+1}| \\
&\leq \lambda^{G(m+2)-G(m+1)} \cdot \lambda^{G(m+3)-G(m+2)} \dots \lambda^{G(\Delta-2)-G(\Delta-3)} |c_{\Delta-3}| \\
&= \lambda^{G(\Delta-2)-G(m+1)} \cdot \lambda^{G(\Delta-1)-G(\Delta-2)} = \lambda^{G(\Delta-1)-G(m+1)}.
\end{aligned}$$

$$
\begin{aligned}
|c_m| &= (\lambda^{G(m+2)-G(m+1)} - 1)|c_{m+1}| + |c_{m+2}| \\
&\geq (\lambda^{G(m+2)-G(m+1)} - 1)|c_{m+1}| \\
&\geq (\lambda^{G(m+2)-G(m+1)} - 1) \cdot (\lambda^{G(m+3)-G(m+2)} - 1) \dots (\lambda^{G(\Delta-2)-G(\Delta-3)} - 1)|c_{\Delta-3}| \\
&= \lambda^{G(\Delta-2)-G(m+1)}|c_{\Delta-3}| \cdot \left(1 - \frac{1}{\lambda^{G(m+2)-G(m+1)}}\right)\left(1 - \frac{1}{\lambda^{G(m+3)-G(m+2)}}\right) \dots \\
&\qquad \dots \left(1 - \frac{1}{\lambda^{G(\Delta-2)-G(\Delta-3)}}\right) \\
&\geq \lambda^{G(\Delta-2)-G(m+1)}|c_{\Delta-3}|. \left(1 - \frac{1}{\lambda^{G(m+2)-G(m+1)}} - \dots - \frac{1}{\lambda^{G(\Delta-2)-G(\Delta-3)}}\right) \\
&\qquad\qquad\qquad\qquad \text{[By Claim 4.1]} \\
&\geq \lambda^{G(\Delta-2)-G(m+1)}|c_{\Delta-3}|. \left(1 - \frac{1}{\lambda^{m+1}} - \frac{1}{\lambda^{m+2}} - \dots - \frac{1}{\lambda^{\Delta-3}}\right) \\
&= \lambda^{G(\Delta-1)-G(m+1)}. \left(1 - \frac{1}{\lambda^m(\lambda-1)}\left(1 - \frac{1}{\lambda^{\Delta-3-m}}\right)\right) \geq \frac{\lambda^{G(\Delta-1)-G(m+1)}}{2}.
\end{aligned}
$$

□

Proof of Lemma 4.1.

The first condition of $(\mathbf{d}, \mathbf{\Delta})$-niceness is satisfied by $(\mathbf{p}, \mathbf{q})$: Indeed we have

$\frac{p}{q} = \frac{c_0}{c_0\lambda - c_1} \implies \frac{1}{2\lambda} \leq \frac{p}{q} \leq \frac{1}{2}$ as $(-c_1)$ is a positive integer less than c_0,

$q \leq |c_0|\lambda + |c_1| \leq 2\lambda^{G(\Delta-1)} \leq d$ where the second inequality follows from the upper bound on each $|c_m|$ in Lemma 4.3.

The second condition of $(\mathbf{d}, \mathbf{\Delta})$-niceness is satisfied by $(\mathbf{p}, \mathbf{q})$: Fix $\delta \in \{2, \cdots, \Delta\}$ and a positive integer $z < \lambda^{G(\delta-1)}/8$. We have to show that

$$\min\left(\frac{zp \bmod q}{q}, 1 - \frac{zp \bmod q}{q}\right) \geq \frac{z}{8\lambda^{G(\delta)-1}} .$$

We will first find what we call the **base $(\mathbf{b_0}, \dots, \mathbf{b_{\Delta-2}})$ representation** of the number z. For $0 \leq m \leq \Delta - 2$, inductively define y_m to be the integer quotient when $\left(z - \sum_{m'=m+1}^{\Delta-2} b_{m'}y_{m'}\right)$ is divided by b_m. Then we can express z as $z = \sum_{m=0}^{\Delta-2} b_m y_m$.

Since $b_m \geq \lambda^{G(m)}/2$ for all m and $z < \lambda^{G(\delta-1)}/8$, we have the following bounds on the values of y_m:

$$y_m = 0 \text{ for } m \geq \delta - 1, \tag{4.2}$$

$$y_{\delta-2} = \left\lfloor \frac{z}{b_{\delta-2}} \right\rfloor < \frac{\frac{\lambda^{G(\delta-1)}}{8}}{\frac{\lambda^{G(\delta-2)}}{2}} \leq \frac{\lambda^{G(\delta-1)-G(\delta-2)} - 1}{2} = \frac{r_{\delta-2}}{2}, \tag{4.3}$$

$$y_m \leq \left\lfloor \frac{b_{m+1} - 1}{b_m} \right\rfloor = r_m \text{ for } m < \delta - 2 \;. \tag{4.4}$$

By (4.1), $zp \equiv \sum\limits_{m=0}^{\Delta-2} c_m y_m \bmod q$. Therefore,

$$\min\left(\frac{zp \bmod q}{q}, 1 - \frac{zp \bmod q}{q}\right) = \min\left(\left|\sum_{m=0}^{\Delta-2} c_m y_m\right|/q,\ 1 - \left|\sum_{m=0}^{\Delta-2} c_m y_m\right|/q\right) \tag{4.5}$$

if $\left|\sum_{m=0}^{\Delta-2} c_m y_m\right|/q \leq 1$, which is true by the following claim (See Section 4.2.3 for the proof):

Claim 4.2. *If* $0 \leq y_m \leq r_m$ *for all* m*, then* $\left|\sum\limits_{m=0}^{\Delta-2} c_m y_m\right| < q - c_0$.

Now let f be the highest index such that $y_f \geq 1$ [by (4.2), $f \leq \delta - 2$] and e be the smallest index such that $y_e \geq 1$. Then $\left|\sum_{m=0}^{\Delta-2} c_m y_m\right| = \left|\sum_{m=e}^{f} c_m y_m\right|$. We need two more claims whose proofs can be found in Section 4.2.3.

Claim 4.3. *Let* y_m *be non-negative integers such that* $y_e \geq 1$*. Then* $\left|\sum_{m=e}^{f} c_m y_m\right| \geq \min\Big(|c_f y_f|, |c_{f-1}| - |c_f y_f|\Big)$.

Claim 4.4. *Let* $\{y_m\}_{m=0}^{\delta-2}$ *be a sequence of non-negative integers. Let* $f \leq \delta - 2$ *be the highest index such that* $y_f \geq 1$*. If* $y_{\delta-2} = \lfloor \frac{z}{b_{\delta-2}} \rfloor \leq r_{\delta-2}/2$ *and* $0 \leq y_m \leq r_m$ *for all* $m \leq \delta - 2$*, then* $\min\Big(|c_f y_f|, |c_{f-1}| - |c_f y_f|\Big) \geq |c_{\delta-2} z/(2b_{\delta-2})|$.

If $\delta = 2$, then $f = 0$ by (4.2). Thus, $q - \left|\sum_{m=e}^{f} c_m y_m\right| > c_0 r_0 - |c_0 y_0| > c_0 r_0/2 > |c_f y_f|$ where the last two inequalities follow from (4.3).

Otherwise $\delta > 2$. By Claim 4.2, $q - \left|\sum_{m=e}^{f} c_m y_m\right| > c_0$. From the definition of the sequence $\{c_m\}$, we have $c_0 \geq |c_f r_f| \geq |c_f y_f|$ when $f > 0$. But when $f = 0$, it follows that $y_{\delta-2} = 0$ implying $z < b_{\delta-2}$. This further implies $c_0 \geq |c_{\delta-2}| \geq |c_{\delta-2} z / b_{\delta-2}|$.

From the analysis of the two cases above and by Claims 4.3 and 4.4, we get that

$$\min\left(\left|\sum_{m=e}^{f} c_m y_m\right|,\ q - \left|\sum_{m=e}^{f} c_m y_m\right|\right)/q \geq \left|\frac{c_{\delta-2} z}{2 b_{\delta-2} q}\right|.$$

By Lemma 4.3, we have

$$|c_{\delta-2}| \geq \lambda^{G(\Delta-1)-G(\delta-1)}/2, \quad b_{\delta-2} \leq \lambda^{G(\delta-2)}, \quad q \leq |c_0|\lambda + |c_1| \leq 2\lambda^{G(\Delta-1)}\ .$$

Hence, $\min\left(\left|\sum_{m=e}^{f} c_m y_m\right|/q,\ 1 - \left|\sum_{m=e}^{f} c_m y_m\right|/q\right) \geq \frac{z}{8\lambda^{G(\delta-1)+G(\delta-2)}} = \frac{z}{8\lambda^{G(\delta)-1}}$ which together with (4.5) implies

$$\min\left(\frac{zp \bmod q}{q}, 1 - \frac{zp \bmod q}{q}\right) \geq \frac{z}{8\lambda^{G(\delta)-1}}\ .$$

□

4.2.3 Missing proofs of technical lemmas

We present the missing proofs of the technical lemmas used in the proof of Lemma 4.1. In the following lemmas, let the sequences $\{b_m\}, \{c_m\}, \{r_m\}$ be as defined in Section 4.2.1.

Claim 4.2. *If* $0 \leq y_m \leq r_m$ *for all* m*, then* $\left|\sum_{m=0}^{\Delta-2} c_m y_m\right| < q - c_0$.

Proof.

$$\sum_{m=0}^{\Delta-2} c_m y_m = \sum_{m=0}^{\lfloor\frac{\Delta-2}{2}\rfloor} c_{2m} y_{2m} + \sum_{m=1}^{\lceil\frac{\Delta-2}{2}\rceil} c_{2m-1} y_{2m-1}$$

where the first summand is ≥ 0 and the second summand is ≤ 0 as c_i takes positive values at even indices and negative values at odd indices. Hence $\left|\sum_{m=0}^{\Delta-2} c_m y_m\right|$ is upper bounded by the maximum of the absolute values of these two summands.

$$\left|\sum_{m=0}^{\lfloor\frac{\Delta-2}{2}\rfloor} c_{2m}y_{2m}\right| \leq \left|\sum_{m=0}^{\lfloor\frac{\Delta-2}{2}\rfloor} c_{2m}r_{2m}\right| = \left|c_0r_0 - c_1 + \left(c_1 + \sum_{m=1}^{\lfloor\frac{\Delta-2}{2}\rfloor} c_{2m}r_{2m}\right)\right|$$

$$\text{and } \left|\sum_{m=1}^{\lceil\frac{\Delta-2}{2}\rceil} c_{2m-1}y_{2m-1}\right| \leq \left|\sum_{m=1}^{\lceil\frac{\Delta-2}{2}\rceil} c_{2m-1}r_{2m-1}\right| = \left|-c_0 + \left(c_0 + \sum_{m=1}^{\lceil\frac{\Delta-2}{2}\rceil} c_{2m-1}r_{2m-1}\right)\right|$$

By repeated substitution of the form $c_m + c_{m+1}r_{m+1} = c_{m+2}$, the first equation becomes equal to $(c_0r_0 - c_1) + c_{2\lfloor\frac{\Delta-2}{2}\rfloor+1}$ and the second equation becomes equal to $\left|-c_0 + c_{2\lceil\frac{\Delta-2}{2}\rceil}\right| = c_0 - c_{2\lceil\frac{\Delta-2}{2}\rceil}$ [We might need to define $c_{\Delta-1} := c_{\Delta-2}r_{\Delta-2} + c_{\Delta-3}$ for this as we have not defined it earlier. It is easy to see that the sign parity of $c_{\Delta-1}$ will be $(-1)^{\Delta-1}$].

Finally,

$$(c_0r_0 - c_1) + c_{2\lfloor\frac{\Delta-2}{2}\rfloor+1} < q - c_0 \qquad \text{as } q - c_0 = c_0r_0 - c_1 \text{ and } c_{2\lfloor\frac{\Delta-2}{2}\rfloor+1} \text{ is negative;}$$

$$c_0 - c_{2\lceil\frac{\Delta-2}{2}\rceil} < q - c_0 \qquad \text{as } q - c_0 = c_0r_0 - c_1 > c_0r_0 > c_0 \text{ and } c_{2\lceil\frac{\Delta-2}{2}\rceil} \text{ is positive.}$$

□

We will need the following lemma for proving Claim 4.3.

Lemma 4.4

Let $z_e, \ldots, z_f$ be integers with $0 \leq z_m \leq r_m$ $\forall m$ and $f \geq e+2$. Also let Y be an integer of the same sign as c_e such that $|Y| \geq |c_e|$. Then there exists an integer Y' of the same sign as c_{e+2} such that $|Y'| \geq |c_{e+2}|$ and

$$|Y + c_ez_e + \sum_{m=e+1}^{f} c_mz_m| = |Y' + c_{e+2}z_{e+2} + \sum_{m=e+3}^{f} c_mz_m|$$

Proof.

$$
\begin{aligned}
&|Y + c_e z_e + \sum_{m=e+1}^{f} c_m z_m| \\
=&|(Y - c_e) + c_e z_e + (c_e + c_{e+1} r_{e+1}) - c_{e+1}(r_{e+1} - z_{e+1}) + \sum_{m=e+2}^{f} c_m z_m| \\
=&|(Y - c_e) + c_e z_e + c_{e+2} - c_{e+1}(r_{e+1} - z_{e+1}) + \sum_{m=e+2}^{f} c_m z_m| \\
=&|Y' + c_{e+2} z_{e+2} + \sum_{m=e+3}^{f} c_m z_m| \qquad \text{where } Y' = (Y - c_e) + c_e z_e + c_{e+2} - c_{e+1}(r_{e+1} - z_{e+1})
\end{aligned}
$$

Each of the terms $(Y - c_e)$, $c_e z_e$, c_{e+2} and $-c_{e+1}(r_{e+1} - z_{e+1})$ is either zero or has the same sign as c_{e+2} because

1. Y and c_e are of the same sign and $|Y| \geq |c_e|$
2. $z_{e+1} \leq r_{e+1}$
3. $c_e, -c_{e+1}$ and c_{e+2} have the same sign

Hence $Y' = (Y - c_e) + c_e z_e + c_{e+2} - c_{e+1}(r_{e+1} - z_{e+1})$ has the same sign as c_{e+2} and

$$|Y'| = |Y - c_e| + |c_e z_e| + |c_{e+2}| + | - c_{e+1}(r_{e+1} - z_{e+1})| \geq |c_{e+2}|.$$

□

Claim 4.3. *Let y_m be non-negative integers such that $y_e \geq 1$. Then* $\left|\sum_{m=e}^{f} c_m y_m\right| \geq \min\Big(|c_f y_f|, |c_{f-1}| - |c_f y_f|\Big)$.

Proof. • If $e = f$, then

$$\left|\sum_{m=e}^{f} c_m y_m\right| = |c_f y_f| \ .$$

- If $e = f - 1$, then

$$\left|\sum_{m=e}^{f} c_m y_m\right| = |c_f y_f + c_{f-1} y_{f-1}| \geq |c_{f-1} y_{f-1}| - |c_f y_f|$$
$$\geq |c_{f-1}| - |c_f y_f| \, . \qquad \text{[because } y_{f-1} = y_e \geq 1\text{]}$$

- If $f - e \geq 2$ and $f - e$ is even, then

$$\left|\sum_{m=e}^{f} c_m y_m\right| = \left|Y + c_e(y_e - 1) + \sum_{m=e+1}^{f} c_m y_m\right| \text{ where } Y = c_e$$
$$= |Y' + c_f y_f| \text{ where } Y' \text{ has the same sign as } c_f$$
$$\text{[By repeated application of Lemma 4.4]}$$
$$\geq |c_f y_f| \, .$$

- If $f - e \geq 2$ and $f - e$ is odd, then

$$\left|\sum_{m=e}^{f} c_m y_m\right| = \left|Y + c_e(y_e - 1) + \sum_{m=e+1}^{f} c_m y_m\right| \text{ where } Y = c_e$$
$$= |Y' + c_{f-1} y_{f-1} + c_f y_f| \text{ where } Y' \text{ has the same sign as } c_{f-1}$$
$$\text{and } |Y'| \geq |c_{f-1}|$$
$$\text{[By repeated application of Lemma 4.4]}$$
$$\geq |Y' + c_{f-1} y_{f-1}| - |c_f y_f|$$
$$\geq |Y'| - |c_f y_f|$$
$$\geq |c_{f-1}| - |c_f y_f| \, .$$

Hence in all four cases, $\left|\sum_{m=e}^{f} c_m y_m\right| \geq \min\left(|c_f y_f|, |c_{f-1}| - |c_f y_f|\right)$. □

Claim 4.4. *Let $\{y_m\}_{m=0}^{\delta-2}$ be a sequence of non-negative integers. Let $f \leq \delta - 2$ be the highest index such that $y_f \geq 1$. If $y_{\delta-2} = \lfloor \frac{z}{b_{\delta-2}} \rfloor \leq r_{\delta-2}/2$ and $0 \leq y_m \leq r_m$ for* *all* $m \leq \delta - 2$, *then* $\min\Big(|c_f y_f|, |c_{f-1}| - |c_f y_f|\Big) \geq |c_{\delta-2} z/(2b_{\delta-2})|$.

Proof. If $f = \delta - 2$ i.e. $y_{\delta-2} \geq 1$, then

$$|c_f y_f| = |c_{\delta-2} y_{\delta-2}| \text{ and}$$
$$|c_{f-1}| - |c_f y_f| = |c_{\delta-3}| - |c_{\delta-2} y_{\delta-2}| \geq |c_{\delta-3}| - \left|c_{\delta-2}\frac{r_{\delta-2}}{2}\right| \geq \left|c_{\delta-2}\frac{r_{\delta-2}}{2}\right| \geq |c_{\delta-2} y_{\delta-2}|$$

where the the second inequality follows from $|c_{\delta-3}| = |c_{\delta-2} r_{\delta-2}| + |c_{\delta-1}|$. As $y_{\delta-2} \geq 1$, we obtain $|c_{\delta-2} y_{\delta-2}| = \left|c_{\delta-2}\left\lfloor \frac{z}{b_{\delta-2}} \right\rfloor\right| \geq \left|\frac{c_{\delta-2} z}{2b_{\delta-2}}\right|$.

Otherwise if $f < \delta - 2$ i.e. $y_{\delta-2} = 0$ i.e. $z < b_{\delta-2}$, then

$$|c_f y_f| \geq |c_f| \geq |c_{\delta-2}| \text{ and}$$
$$|c_{f-1}| - |c_f y_f| \geq |c_{f-1}| - |c_f r_f| = |c_{f+1}| \geq |c_{\delta-2}|$$

where the last inequality on each of the above two lines follows from $f < \delta - 2$ and the fact that $|c_m|$ decreases as m increases. As $z < b_{\delta-2}$, we get $|c_{\delta-2}| > \left|\frac{c_{\delta-2} z}{b_{\delta-2}}\right|$.

Hence in both the cases, $\min\Big(|c_f y_f|, |c_{f-1}| - |c_f y_f|\Big) \geq |c_{\delta-2} z/(2b_{\delta-2})|$. □

Bibliography

[Ajt83] M. Ajtai. Σ_1^1-formulae on finite structures. *Annals of Pure and Applied Logic*, 24(1):1, 1983. 4

[AV08] Manindra Agrawal and V. Vinay. Arithmetic Circuits: A Chasm at Depth Four. In *2008 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 67–75, 2008. 4

[BCS97] Peter Bürgisser, Michael Clausen, and M. Amin Shokrollahi. *Algebraic complexity theory*, volume 315 of *Grundlehren der mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]*. Springer-Verlag, Berlin, 1997. With the collaboration of Thomas Lickteig.

[BDS22] C. S. Bhargav, Sagnik Dutta, and Nitin Saxena. Improved Lower Bound, and Proof Barrier, for Constant Depth Algebraic Circuits. In Stefan Szeider, Robert Ganian, and Alexandra Silva, editors, *47th International Symposium on Mathematical Foundations of Computer Science (MFCS 2022)*, volume 241 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 18:1–18:16, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. vi

[BS83] Walter Baur and Volker Strassen. The complexity of partial derivatives. *Theoret. Comput. Sci.*, 22(3):317–330, 1983.

[CKW10] Xi Chen, Neeraj Kayal, and Avi Wigderson. Partial derivatives in arithmetic complexity and beyond. *Found. Trends Theor. Comput. Sci.*, 6(1-2):front matter, 1–138 (2011), 2010.

[CLS19] Suryajith Chillara, Nutan Limaye, and Srikanth Srinivasan. Small-depth multilinear formula lower bounds for iterated matrix multiplication with applications. *SIAM J. Comput.*, 48(1):70–92, 2019.

[FLMS15] Hervé Fournier, Nutan Limaye, Guillaume Malod, and Srikanth Srinivasan. Lower bounds for depth-4 formulas computing iterated matrix multiplication. *SIAM J. Comput.*, 44(5):1173–1201, 2015.

[FSS81] Merrick Furst, James B. Saxe, and Michael Sipser. Parity, circuits, and the polynomial-time hierarchy. In *22nd Annual Symposium on Foundations of Computer Science (sfcs 1981)*, pages 260–270, 1981. 4

[GKKS14] Ankit Gupta, Pritish Kamath, Neeraj Kayal, and Ramprasad Saptharishi. Approaching the chasm at depth four. *J. ACM*, 61(6):Art. 33, 16, 2014.

[GKKS16] Ankit Gupta, Pritish Kamath, Neeraj Kayal, and Ramprasad Saptharishi. Arithmetic circuits: a chasm at depth 3. *SIAM J. Comput.*, 45(3):1064–1079, 2016. 4

[GST20] Nikhil Gupta, Chandan Saha, and Bhargav Thankey. A super-quadratic lower bound for depth four arithmetic circuits. In *35th Computational Complexity Conference*, volume 169 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. 23, 31. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2020. 4

[Has86] J Hastad. Almost optimal lower bounds for small depth circuits. In *Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing*, STOC '86, page 6–20, New York, NY, USA, 1986. Association for Computing Machinery. 4

[Kal85] K. A. Kalorkoti. A lower bound for the formula size of rational functions. *SIAM J. Comput.*, 14(3):678–687, 1985.

[Kay12] Neeraj Kayal. An exponential lower bound for the sum of powers of bounded degree polynomials. 2012.

[KLSS17] Neeraj Kayal, Nutan Limaye, Chandan Saha, and Srikanth Srinivasan. An exponential lower bound for homogeneous depth four arithmetic formulas. *SIAM J. Comput.*, 46(1):307–335, 2017.

[Koi12] Pascal Koiran. Arithmetic circuits: the chasm at depth four gets wider. *Theoret. Comput. Sci.*, 448:56–65, 2012. 4

[KS] Deepanshu Kush and Shubhangi Saraf. Improved Low-Depth Set-Multilinear Circuit Lower Bounds. to appear in CCC 2022. 5

[KS14] Mrinal Kumar and Shubhangi Saraf. On the power of homogeneous depth 4 arithmetic circuits. In *55th Annual IEEE Symposium on Foundations of Computer Science—FOCS 2014*, pages 364–373. IEEE Computer Soc., Los Alamitos, CA, 2014.

[KS15] Mrinal Kumar and Shubhangi Saraf. The limits of depth reduction for arithmetic formulas: it's all about the top fan-in. *SIAM J. Comput.*, 44(6):1601–1625, 2015.

[KS23] Deepanshu Kush and Shubhangi Saraf. Near-optimal set-multilinear formula lower bounds. *Electron. Colloquium Comput. Complex.*, TR23-017, 2023. 6

[KSS14] Neeraj Kayal, Chandan Saha, and Ramprasad Saptharishi. A super-polynomial lower bound for regular arithmetic formulas. In *Symposium on Theory of Computing (STOC)*. ACM - Association for Computing Machinery, June 2014.

[KST16] Neeraj Kayal, Chandan Saha, and Sébastien Tavenas. An almost cubic lower bound for depth three arithmetic circuits. In *43rd International Colloquium on Automata, Languages, and Programming*, volume 55 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 33, 15. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2016. 4

[KST18] Neeraj Kayal, Chandan Saha, and Sébastien Tavenas. On the size of homogeneous and of depth-four formulas with low individual degree. *Theory Comput.*, 14:Paper No. 16, 46, 2018.

[LST] Nutan Limaye, Srikanth Srinivasan, and Sébastien Tavenas. Superpolynomial Lower Bounds Against Low-Depth Algebraic Circuits. to appear in FOCS, 2021. iv, 4, 5, 6, 7, 8, 9, 10, 11, 12, 16

[Mah14] Meena Mahajan. Algebraic complexity classes. In *Perspectives in computational complexity*, volume 26 of *Progr. Comput. Sci. Appl. Logic*, pages 51–75. Birkhäuser/Springer, Cham, 2014.

[Mit70] D. S. Mitrinović. *Analytic inequalities*. Die Grundlehren der mathematischen Wissenschaften, Band 165. Springer-Verlag, New York-Berlin, 1970. In cooperation with P. M. Vasić. 23

[NW95] Noam Nisan and Avi Wigderson. Lower bounds on arithmetic circuits via partial derivatives. *Comput. Complexity*, 6(3):217–234, 1995. 5

[Raz87] Alexander A. Razborov. Lower bounds on the size of bounded depth circuits over a complete basis with logical addition. *Mathematical notes of the Academy of Sciences of the USSR*, 41:333–338, 1987. 4

[Raz06] Ran Raz. Separation of multilinear circuit and formula size. *Theory Comput.*, 2:121–135, 2006.

[Raz09] Ran Raz. Multi-linear formulas for permanent and determinant are of super-polynomial size. *J. ACM*, 56(2):Art. 8, 17, 2009.

[Raz10] Ran Raz. Elusive functions and lower bounds for arithmetic circuits. *Theory Comput.*, 6:135–177, 2010. 4

[RY09] Ran Raz and Amir Yehudayoff. Lower bounds and separations for constant depth multilinear circuits. *Comput. Complexity*, 18(2):171–207, 2009.

[Sap15] Ramprasad Saptharishi. A survey of lower bounds in arithmetic circuit complexity. *Github Survey*, 2015.

[Sch91] Wolfgang M. Schmidt. *Diophantine approximations and Diophantine equations*, volume 1467 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 1991.

[Smo87] Roman Smolensky. Algebraic methods in the theory of lower bounds for boolean circuit complexity. *Proceedings of the nineteenth annual ACM symposium on Theory of computing*, 1987. 4

[Smo90] R. Smolensky. On interpolation by analytic functions with special properties and some weak lower bounds on the size of circuits with symmetric gates. In *Proceedings [1990] 31st Annual Symposium on Foundations of Computer Science*, pages 628–631 vol.2, 1990.

[SS97] Victor Shoup and Roman Smolensky. Lower bounds for polynomial evaluation and interpolation problems. *Comput. Complexity*, 6(4):301–311, 1996/97. 4

[SW01] Amir Shpilka and Avi Wigderson. Depth-3 arithmetic circuits over fields of characteristic zero. *Comput. Complexity*, 10(1):1–27, 2001.

[SY09] Amir Shpilka and Amir Yehudayoff. Arithmetic circuits: a survey of recent results and open questions. *Found. Trends Theor. Comput. Sci.*, 5(3-4):207–388 (2010), 2009.

[Tav13] Sébastien Tavenas. Improved bounds for reduction to depth 4 and depth 3. In *Mathematical foundations of computer science 2013*, volume 8087 of *Lecture Notes in Comput. Sci.*, pages 813–824. Springer, Heidelberg, 2013. 4

[TLS] Sébastien Tavenas, Nutan Limaye, and Srikanth Srinivasan. Set-multilinear and non-commutative formula lower bounds for iterated matrix multiplication. to appear in STOC 2022. 5

[TSL] Sébastien Tavenas, Srikanth Srinivasan, and Nutan Limaye. On the Partial Derivative Method Applied to Lopsided Set-Multilinear Polynomials. to appear in CCC 2022.

[Val79] L. G. Valiant. Completeness classes in algebra. In *Conference Record of the Eleventh Annual ACM Symposium on Theory of Computing (Atlanta, Ga., 1979)*, pages 249–261. ACM, New York, 1979. 3

[VSBR83] L. G. Valiant, S. Skyum, S. Berkowitz, and C. Rackoff. Fast parallel computation of polynomials using few processors. *SIAM J. Comput.*, 12(4):641–644, 1983. 4