A Super-Quadratic Lower Bound for Depth Four Arithmetic Circuits

Nikhil Gupta
Department of Computer Science and Automation, Indian Institute of Science, Bangalore, India
nikhilg@iisc.ac.in

Chandan Saha
Department of Computer Science and Automation, Indian Institute of Science, Bangalore, India
chandan@iisc.ac.in

Bhargav Thankey
Department of Computer Science and Automation, Indian Institute of Science, Bangalore, India
thankeyd@iisc.ac.in

Abstract

We show an $\widetilde{\Omega}(n^{2.5})$ lower bound for general depth four arithmetic circuits computing an explicit n-variate degree-$\Theta(n)$ multilinear polynomial over any field of characteristic zero. To our knowledge, and as stated in the survey [88], no super-quadratic lower bound was known for depth four circuits over fields of characteristic $\neq 2$ before this work. The previous best lower bound is $\widetilde{\Omega}(n^{1.5})$ [85], which is a slight quantitative improvement over the roughly $\Omega(n^{1.33})$ bound obtained by invoking the super-linear lower bound for constant depth circuits in [73, 86].

Our lower bound proof follows the approach of the almost cubic lower bound for depth three circuits in [53] by replacing the shifted partials measure with a suitable variant of the projected shifted partials measure, but it differs from [53]'s proof at a crucial step – namely, the way "heavy" product gates are handled. Loosely speaking, a heavy product gate has a relatively high fan-in. Product gates of a depth three circuit compute products of affine forms, and so, it is easy to prune $\Theta(n)$ many heavy product gates by projecting the circuit to a low-dimensional affine subspace [53, 87]. However, in a depth four circuit, the second (from the top) layer of product gates compute products of polynomials having *arbitrary* degree, and hence it was not clear how to prune such heavy product gates from the circuit. We show that heavy product gates can also be eliminated from a depth four circuit by projecting the circuit to a low-dimensional affine subspace, unless the heavy gates together account for $\widetilde{\Omega}(n^{2.5})$ size. This part of our argument is inspired by a well-known greedy approximation algorithm for the weighted set-cover problem.

2012 ACM Subject Classification Theory of computation → Algebraic complexity theory

Keywords and phrases depth four arithmetic circuits, Projected Shifted Partials, super-quadratic lower bound

Digital Object Identifier 10.4230/LIPIcs.CCC.2020.23

Acknowledgements We would like to thank Neeraj Kayal and Ankit Garg for sitting through a presentation of this work and giving us useful feedback. Thanks specially to Ankit for bringing the work of Chen and Tell [20] to our notice. A part of this work is done at Microsoft Research India (MSRI), where CS is spending a sabbatical year. CS would like to thank MSRI for providing an excellent research environment and for the hospitality.

1 Introduction

The arithmetic circuit model is naturally well-suited for the study of optimality of algorithms for algebraic and linear algebraic problems. An arithmetic circuit consists of addition ($+$) and multiplication ($\times$) gates, it takes input $\{x_1, x_2, \ldots, x_n\}$ and field scalars, and computes a polynomial in $\{x_1, x_2, \ldots, x_n\}$. *Size* of a circuit is the number of wires in it, and *depth*

35th Computational Complexity Conference (CCC 2020).
Editor: Shubhangi Saraf; Article No. 23; pp. 23:1–23:31
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

is the longest path from an input to an output gate. The two complexity measures – size and depth – of a circuit essentially capture the sequential and parallel complexity of the computation happening inside the circuit.

Arithmetic circuits are *weaker*[1] than Boolean circuits: An explicit lower bound on the size of Boolean circuits implies an explicit lower bound on the size of arithmetic circuits over finite fields[2] and also over fields of characteristic zero[3], but the converse is not true[4]. Still, the best known lower bound for arithmetic circuits is $\Omega(n \log n)$ [13, 90], which is barely super-linear. For arithmetic formulas, an $\Omega(n^2)$ lower bound is known [44]. Several other better lower bounds have been shown in the past few decades for various restricted models of arithmetic circuits (see Section A for a brief history of known lower bounds). But, very few lower bounds are known for circuit models that do not impose restrictions like homogeneity, multilinearity, monotonicity, bounded coefficients, bounded reads etc. One such result is the lower bound for constant depth circuits.

Shoup and Smolensky [86], and Raz [73], showed an $\Omega(\Delta n^{1+\frac{1}{\Delta}})$ lower bound for depth-Δ circuits, where $\Delta = O(\log n)$. In the special case of depth three circuits, an $\Omega(n^2)$ lower bound was shown in [87], which was improved to an $\widetilde{\Omega}(n^3)$ lower bound in [53]. However, for depth four circuits[5], the best lower bound was $\widetilde{\Omega}(n^{1.5})$ [85], which is a slight quantitative improvement over the roughly $\Omega(n^{1.33})$ lower bound obtained by specializing the constant depth lower bound in [73, 86] to depth four circuits.

In this work, we show an $\widetilde{\Omega}(n^{2.5})$ lower bound for depth four circuits. To the best of our knowledge, and as stated in the survey [88] (Section 1.4.2, page-13), this is the first super-quadratic lower bound for this model over fields of characteristic $\neq 2$.

1.1 Our Result

We state our result formally now. Without loss of generality, we will assume that a depth four circuit is a $\Sigma\Pi\Sigma\Pi$ circuit, i.e., the circuit has a $+$-gate on top followed by second layer of $\times$-gates, then a third layer of $+$-gates and finally a bottom layer of $\times$-gates.

▶ **Theorem 1** (Lower bound for depth four circuits). *Over any field of characteristic zero*[6]*, there exists a family of mulitilinear polynomials $\{f_n\}_{n\geq 1}$ in VNP, where f_n is a polynomial in $\Theta(n)$ variables and of degree $\Theta(n)$ such that any depth four circuit computing f_n has $\Omega\left(\frac{n^{2.5}}{(\log n)^6}\right)$ many wires/edges and $\Omega\left(\frac{n^{1.5}}{(\log n)^4}\right)$ many gates.*

---

[1] This is not to mean that arithmetic circuits cannot simulate Boolean circuits. It is easy to see that a Boolean circuit can be efficiently simulated by an arithmetic circuit over any field that contains the additive and the multiplicative identities 0 and 1 respectively.

[2] This is because an arithmetic circuit over a finite field can be simulated by a Boolean circuit with only a slight blow-up in size and depth (see [100]).

[3] It was shown in [16] (Corollary 4.6 in Chapter 4) that $\mathsf{NC}^3/\text{poly} \neq \mathsf{NP}/\text{poly}$ implies $\mathsf{VP} \neq \mathsf{VNP}$ over fields of characteristic zero, assuming the Generalized Riemann Hypothesis. The circuit classes VP and VNP are arithmetic analogues of non-uniform P and NP respectively [96].

[4] as computing a Boolean function is a weaker requirement than computing a specific polynomial representation of the function.

[5] Depth four circuits form a natural circuit class as the "optimal" circuit for an arbitrary polynomial turns out to be a depth four circuit: The *multiplicative complexity* of a polynomial f, denoted $M(f)$, is the minimum number of multiplication gates required to compute f. It is known that there exists an n-variate degree-d polynomial f for which $M(f) = \Omega(\sqrt{\binom{n+d}{d}})$ [21, 39]. On the other hand, every n-variate degree-d polynomial can be computed by a depth four circuit having $\sqrt{\binom{n+d}{d}} \cdot \text{poly}(n, d)$ many multiplication gates [61].

[6] The lower bound holds even if the characteristic is sufficiently large (see Section 4).

A word about the polynomial family. The polynomial family $\{f_n\}_{n\geq 1}$ is a variant of the Nisan-Wigderson design polynomial family, which has been used in proving several other previous lower bounds [49, 51–53, 57, 58, 73, 85]. The n-th member of the family, i.e., f_n is a polynomial in $\mathbf{x} = \{x_1, ..., x_{3m}\}$ and $\mathbf{y} = \{y_1, ..., y_{3m}\}$ variables, where m is an integer in $\left[\frac{n}{2}, 2n\right]$. The degree of the polynomial in $\mathbf{y}$ variables is $\deg_{\mathbf{y}}(f_n) = m$, while its degree in $\mathbf{x}$ variables is $\deg_{\mathbf{x}}(f_n) = d_{\mathbf{x}} = \Theta(\frac{\sqrt{m}}{\ln m})$. Informally, f_n contains multiple 'copies' of the design polynomial in different subsets of the $\mathbf{x}$ variables, while the $\mathbf{y}$ variables are used as 'prefixes' to uniquely identify each such copy. While the reason for having multiple copies is similar to [53], as we shall see in the next section, handling them is a little trickier in our case. Note that because of the way we have defined m and $d_{\mathbf{x}}$, proving that any depth four circuit computing f_n has $\Omega\left(\frac{m^2 d_{\mathbf{x}}}{(\ln m)^5}\right)$ many edges and $\Omega\left(\frac{m d_{\mathbf{x}}}{(\ln m)^3}\right)$ many gates would establish the theorem.[7] The exact description of f_n is given in Section 4.

1.2 Proof Idea

Let $\mathtt{C}$ be a depth four circuit over a field $\mathbb{F}$. Like many other works on arithmetic circuit lower bounds, we use a rank based complexity measure to obtain our result. The measure we apply is a variant of the *projected shifted partials* measure, which has been used before in [49, 51, 58, 85] and other works. Our proof can be divided into four steps; the first three show a "small" upper bound on the measure of $\mathtt{C}$ while the last step shows a "large" lower bound on the measure of the hard polynomial f_n described above. We now briefly outline each of these steps.

Step 1: Restricting the bottom support of $\mathtt{C}$. We begin by removing all monomials computed by the bottom layer of $\mathtt{C}$ that have a "large" support. Such restrictions have been used in previous works [49, 51, 56, 58, 85] to control the degree of the (sparse) polynomials computed at the third layer of a depth four circuit so that a "small" upper bound on the measure of the circuit can be obtained. However, in our work, this step plays an even more significant role by enabling us to remove "heavy" gates (see below). While previous works use *random* restrictions, we use a *deterministic* procedure for restricting the bottom support – we briefly explain our reasons for doing so towards the end of this section.

Heavy gates. We call a product gate in the second layer of $\mathtt{C}$ *heavy* if the number of distinct third layer gates (computing sparse polynomials) feeding into it is $\widetilde{\Omega}(n^{1.5})$. The presence of heavy gates makes the task of obtaining a "small" upper bound on the measure of $\mathtt{C}$ difficult. The problem of dealing with heavy gates was also faced by previous works on depth three circuits [53, 87], and was dealt with by removing all heavy gates from the circuit before applying the measure to it. We too remove all heavy gates from $\mathtt{C}$, but our way of doing so differs from [53, 87]. Since a depth three circuit computes a sum of product of affine forms, [53, 87] were able to remove all heavy gates by going modulo affine factors of these gates thereby restricting the circuit to an affine subspace. While going modulo the sparse factors of heavy gates is a natural generalization of this technique for depth four circuits, we do not know how to adopt this method as the quotient ring so obtained might not be a polynomial ring. In the next step, we outline a technique of removing heavy gates from $\mathtt{C}$ which, in spirit, also restricts $\mathtt{C}$ to an affine subspace.

[7] We use log base e in the proof rather than log base 2 as it simplifies the analysis.

Step 2: Removing heavy gates from $\mathtt{C}$. We remove heavy gates from $\mathtt{C}$ by sequentially evaluating *exactly one* sparse factor of each heavy gate to zero. This can be done if $\mathbb{F}$ is algebraically closed, which one can assume without loss of generality (as argued in Section 5). In particular, we use the following greedy procedure: While there exists a heavy gate in $\mathtt{C}$, pick a sparse polynomial for which the ratio of the number of heavy gates connected to it to its fan-in is maximum, and evaluate it to zero. Intuitively, this allows us to remove a large number of heavy gates at the cost of evaluating a few monomials (computed by the bottom layer) to field constants. Then, as we have restricted the bottom support of $\mathtt{C}$ in Step 1, we are able to show that we can remove $\Theta(n)$ many heavy gates at a cost of setting only a few variables to constants (unless the already removed heavy gates account for $\widetilde{\Omega}(n^{2.5})$ size). Note that $\Theta(n)$ many heavy gates would immediately imply the desired lower bound. This greedy procedure is inspired by an approximation algorithm for the weighted set cover problem [99] (Section 2.1, page-16), however its analysis here is tailored to our needs. This step, which is the main contribution of this work, plays a crucial role in enabling us to prove a super-quadratic lower bound and we provide more details about it in Section 3.2.2.

Step 3: Analyzing the measure of $\mathtt{C}$. After all the heavy gates have been removed, a "small" upper bound on the measure of $\mathtt{C}$ is derived by closely following the "grouping" argument made in [53]. However, we replace the shifted partials measure by the projected shifted partials measure as the latter is suitable for controlling the degree of the sparse polynomials computed at the third layer of $\mathtt{C}$. In this step, we divide the factors of a polynomial T computed by a $\times$-gate in the second layer of $\mathtt{C}$ into "groups" of suitable sizes, and multiply out factors in the same group to reduce the effective number of factors of T. This then helps obtain a "small" upper bound on the projected shifted partials measure of T which, by sub-additivity, implies a "small" upper bound on the projected shifted partials measure of $\mathtt{C}$.

Step 4: Lower bound on the measure of f_n. The choice of the hard polynomial f_n is dictated by the above technique of removing heavy gates. As mentioned before, f_n has multiple copies of the Nisan-Wigderson design polynomial, denoted NW. The reason for having multiple copies is that if we work with only one copy, we might end up irreparably damaging it while removing heavy gates from $\mathtt{C}$. On the other hand, starting with multiple copies of NW, much like in [53], we are able to show that the procedure for removing heavy gates leaves an intact copy along with some 'damaged' parts from the other copies. Our use of a deterministic restriction in Step 1 makes it easier to show this. A "large" lower bound on the measure of NW was obtained in [49, 51, 58]. However, it is not clear how to obtain such a lower bound on NW in the presence of other "damaged" parts, and so we remove these parts. Although in [53], such parts were removed by simply setting a subset of variables to 0 and 1, here we need to augment the projected shifted measure appropriately to get rid of them.

2 Preliminaries

Notations. For $r \in \mathbb{N}$, $[r] := \{1, \ldots, r\}$. We use lowercase Greek alphabets like α, β for field constants, bold-face lowercase letters like $\mathbf{x}$ and $\mathbf{y}$ to denote sets of variables, f, g for polynomials in $\mathbb{F}[\mathbf{x}, \mathbf{y}]$, uppercase typewriter alphabets $\mathtt{C}, \mathtt{D}$ for arithmetic circuits over $\mathbb{F}$, uppercase Roman alphabets T, Q for the polynomials computed by the gates of a depth four circuit and M, M_1, M_2 for subsets of natural numbers. For $M \subseteq [3m]$, $\mathbf{x}_M := \{x_i : i \in M\}, \mathbf{y}_M := \{y_i : i \in M\}$. For $\mathbf{z} \subseteq \mathbf{x}_M \cup \mathbf{y}_M$ and $r \in \mathbb{N}$, $\mathbf{z}^{\leq \infty}$ and $\mathbf{z}^{\leq r}$ denote the set of all monomials in $\mathbf{z}$ variables and the set of all monomials in $\mathbf{z}$ variables with degree at most r respectively. For $S \subseteq \mathbb{F}[\mathbf{x}, \mathbf{y}]$, $\dim\langle S \rangle$ denotes the dimension of the $\mathbb{F}$-linear span of S.

Support and degree of a monomial. The support of a monomial η, denoted $\mathrm{Supp}(\eta)$, is the set of variables appearing in it. Also, for any $\mathbf{z} \subseteq \mathbf{x} \cup \mathbf{y}$ we will use $\deg_{\mathbf{z}}(\eta)$ to denote its degree in $\mathbf{z}$ variables. We will say that η is $\mathbf{z}$-multilinear if the degree of every $\mathbf{z}$ variable in η is at most one.

2.1 The complexity measure

Throughout this section, we will assume that $m \in \mathbb{N}$ is as stated in the paragraph following Theorem 1, $M \subseteq [3m]$, $|M| = m$, $f \in \mathbb{F}[\mathbf{x}_M, \mathbf{y}_M]$ and $S \subseteq \mathbb{F}[\mathbf{x}_M, \mathbf{y}_M]$. Note that the set M is not fixed and will depend on the circuit under analysis. Before defining the measure, let us define the operations that make up the measure.

1. **Partial derivatives.** Let $\eta = x_1 \cdots x_k$ be a monomial in $\mathbf{x}$ variables. Then, we define the partial derivative of f with respect to η as

$$\frac{\partial f}{\partial \eta} := \frac{\partial}{\partial x_1}\left(\frac{\partial}{\partial x_2}\left(\cdots\left(\frac{\partial f}{\partial x_k}\right)\right)\right).$$

   If the degree of η is k, then $\frac{\partial f}{\partial \eta}$ is said to be a k-th order partial derivative of f. We denote by $\partial^k_{\mathbf{x}} f$ the set of all k-th order partial derivatives of f taken with respect to multilinear monomials in $\mathbf{x}$ variables.
2. **The shift operation.** Let η be a degree ℓ multilinear monomial in $\mathbf{x}_M$ variables. We say that the polynomial $\eta \cdot f$ is obtained by *shifting* f by η. We denote by $\mathbf{x}^\ell_M f$ the set of polynomials obtained by shifting f by all degree ℓ multilinear monomials in $\mathbf{x}_M$ variables and $\mathbf{x}^\ell_M S := \left\{\mathbf{x}^\ell_M f : f \in S\right\}$.
3. **Multilinear projection.** We define a map $\pi_{\mathbf{x}} : \mathbb{F}[\mathbf{x}_M, \mathbf{y}_M] \to \mathbb{F}[\mathbf{x}_M, \mathbf{y}_M]$ with $\pi_{\mathbf{x}}(f)$ being the polynomial made up of exactly the $\mathbf{x}$-multilinear monomials of f. Formally, for a monomial η, $\pi_{\mathbf{x}}(\eta) = \eta$ if η is $\mathbf{x}$-multilinear and 0 otherwise. The map is then linearly extended for arbitrary polynomials and $\pi_{\mathbf{x}}(S) := \{\pi_{\mathbf{x}}(f) : f \in S\}$.
4. **A degree based projection.** For $i \in \mathbb{N}$ and $f \in \mathbb{F}[\mathbf{x}_M, \mathbf{y}_M]$, we define $[f]_i$ to be the polynomial made up of only those monomials of f whose $\mathbf{y}$-*degree* is exactly i. Formally, for a monomial η, $[\eta]_i = \eta$ if $\deg_{\mathbf{y}}(\eta) = i$ and 0 otherwise. It is then linearly extended for arbitrary polynomials and $[S]_m := \{[f]_m : f \in S\}$.
5. **An evaluation map.** For $\alpha \in \mathbb{F}$ and $\mathbf{z} \subseteq \mathbf{x}_M \cup \mathbf{y}_M$, we define a map $\sigma_{\mathbf{z}=\alpha} : \mathbb{F}[\mathbf{x}_M, \mathbf{y}_M] \to \mathbb{F}[\mathbf{x}_M \setminus \mathbf{z},\ \mathbf{y}_M \setminus \mathbf{z}]$ with $\sigma_{\mathbf{z}=\alpha}(f)$ being obtained from f by setting every variable in $\mathbf{z}$ to α and $\sigma_{\mathbf{z}=\alpha}(S) := \{\sigma_{\mathbf{z}=\alpha}(f) : f \in S\}$.

The operations given in 1, 2 and 3 constitute the projected shifted partials measure [49]. In this work, we define and use the measure $\mathsf{PSP}_{M,k,\ell}$, which is obtained by augmenting the projected shifted partials measure with the operations in 4 and 5 as follows.

▶ **Definition 2** (The measure). *For $m, k, \ell \in \mathbb{N}$, $M \subseteq [3m], |M| = m$ and $f \in \mathbb{F}[\mathbf{x}_M, \mathbf{y}_M]$,*

$$\mathsf{PSP}_{M,k,\ell}(f) := \dim\left\langle \sigma_{\mathbf{y}_M=1}\left(\left[\pi_{\mathbf{x}}\left(\mathbf{x}^\ell_M\ \partial^k_{\mathbf{x}} f\right)\right]_m\right)\right\rangle.$$

▶ **Observation 3** (Sub-additivity of the measure). *For any two polynomials $f, g \in \mathbb{F}[\mathbf{x}_M, \mathbf{y}_M]$,*

$$\mathsf{PSP}_{M,k,\ell}(f+g) \leq \mathsf{PSP}_{M,k,\ell}(f) + \mathsf{PSP}_{M,k,\ell}(g).$$

The above observation is easy to prove and we omit its proof here.

2.2 Some numerical estimates

▶ **Proposition 4** (Estimating Binomial Coefficients). *For any $n, k \in \mathbb{N}$, $k \leq n$, $\left(\frac{n}{k}\right)^k \leq \binom{n}{k} < \left(\frac{en}{k}\right)^k$.*

▶ **Proposition 5** ([33, 49]). *Let $a(n), f(n), g(n) : \mathbb{Z}_{>0} \to \mathbb{Z}$ be integer values functions such that $(|f| + |g|) = o(a)$. Then, $\ln \frac{(a+f)!}{(a-g)!} = (f+g)\ln(a) \pm O\left(\frac{f^2+g^2}{a}\right)$.*

3 Upper bounding the measure for a depth four circuit

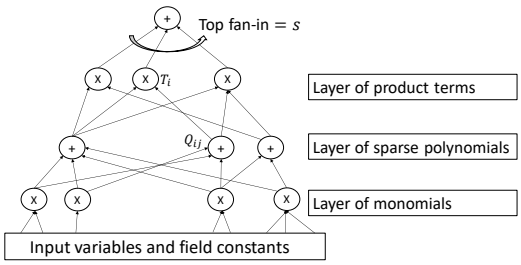
Let $\mathtt{C} = \sum_{i=1}^{s} T_i$ be a depth four circuit computing the polynomial $f = f_n$ (recall $\deg_{\mathbf{x}}(f_n) = d_{\mathbf{x}} = \Theta\left(\frac{\sqrt{m}}{\ln m}\right)$ from Section 1.1); $T_i = \prod_{j=1}^{a_i} Q_{ij}^{e_j}$ and Q_{ij}'s are distinct sparse polynomials computed by the $+$-gates in the third layer of $\mathtt{C}$, and $e_j \geq 1$. We assume that $\mathbb{F}$ is algebraically closed and argue in Section 5 why this holds without loss of generality. For brevity, we would use the terminologies 'product terms', 'sparse polynomials' and 'monomials' for the $\times$-gates, $+$-gates and $\times$-gates in the second, third and fourth layers of $\mathtt{C}$ respectively as shown in Figure 1a. The proof of the upper bound is divided into three steps:

Step 1: Restricting the bottom support. In this step, we show that if $\mathtt{C}$ has fewer than $\Omega\left(\frac{m^2 d_{\mathbf{x}}}{(\ln m)^5}\right)$ distinct monomials then we can remove all monomials with support more than $\tau = \lfloor 20 \ln m \rfloor$ at a cost of setting m many $\mathbf{x}$ variables and m many $\mathbf{y}$ variables to zero. (Notice that if there are more than $\Omega\left(\frac{m^2 d_{\mathbf{x}}}{(\ln m)^5}\right)$ many monomials then there is nothing to prove.) This step is required not only to remove heavy gates in Step 2 but also in Step 3 where using the fact that all monomials have support at most τ and multilinear projection, we will argue that the degree of all monomials is not too large. More details about this restriction are given in Section 3.2.1.

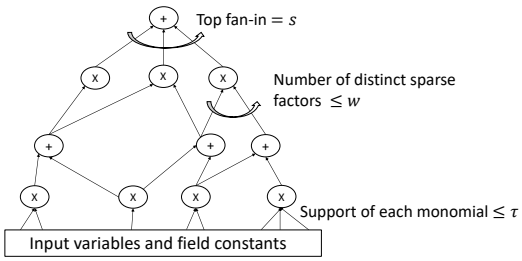
Step 2: Removing heavy gates. The transformed circuit $\mathtt{C}_1$, obtained after Step 1, computes a polynomial in the remaining $2m$ many $\mathbf{x}$ variables and $2m$ many $\mathbf{y}$ variables. Moreover, the number of gates and the fan-in of all gates in $\mathtt{C}_1$ is upper bounded by the number of gates and the fan-in of the corresponding gates in $\mathtt{C}$. A product term in $\mathtt{C}_1$ is called a *heavy gate* if at least $w = \left\lfloor \frac{m d_{\mathbf{x}}}{\lambda_0 \cdot (\ln m)^3} \right\rfloor$ (λ_0 is a large enough constant fixed in Appendix C) many distinct sparse polynomials are connected to it. If there are more than m heavy gates, we are done. Otherwise, we remove all heavy gates using the following greedy procedure: While there is a heavy gate, evaluate a sparse polynomial that would kill the most number of heavy gates at the cost of evaluating as few monomials as possible. As we have already restricted the support of all monomials to τ, we are able to argue in Section 3.2.2 that this can be done at a cost of setting m many $\mathbf{x}$ and m many $\mathbf{y}$ variables to field constants.

These steps transform $\mathtt{C}$ to a 'pruned circuit', defined as follows and depicted in Figure 1b.

▶ **Definition 6.** *We say that a depth four circuit $\mathtt{D}$ is a* pruned circuit *if the support of all monomials in $\mathtt{D}$ is at most $\tau = \lfloor 20 \ln m \rfloor$, and it does not contain any heavy gate; i.e. the number of distinct sparse polynomials feeding into any product term in $\mathtt{D}$ is less than $w = \left\lfloor \frac{m d_{\mathbf{x}}}{\lambda_0 \cdot (\ln m)^3} \right\rfloor$.*



(a) A depth four circuit C.



(b) The pruned depth four circuit D.

Figure 1 A depth four circuit and its pruned version.

Step 3: Analysing the measure. In this step, we analyse the measure $\mathsf{PSP}_{M,k,\ell}$ of the pruned circuit D, obtained after Steps 1 and 2, computing a polynomial in the remaining m many $\mathbf{x}$ and m many $\mathbf{y}$ variables. More details on this analysis are provided in the following section.

3.1 Upper bound on the measure of a pruned depth four circuit

Recall the definition of the measure $\mathsf{PSP}_{M,k,\ell}$ from Section 2. In Steps 1 and 2, we will ensure that if a variable x_i is set to a field constant then y_i is also set to a field constant and vice versa. The set M is then the set of indices of the remaining $\mathbf{x}$ (or $\mathbf{y}$) variables and $|M| = m$.

▶ **Lemma 7.** *Let D be a pruned depth four circuit with top fan-in s computing a polynomial in $\mathbf{x}_M$ and $\mathbf{y}_M$ variables, where $M \subseteq [3m], |M| = m$. Also, let $d_{\mathbf{x}}, \tau, w$ be as defined earlier, $t = \left\lfloor \frac{d_{\mathbf{x}}}{(\ln m)^3} \right\rfloor$, $\delta = \frac{1}{(\ln m)^2}$, $k = \left\lfloor \frac{\delta d_{\mathbf{x}}}{t} \right\rfloor$ and $\ell = \left\lfloor \frac{m}{m^{\delta/t}+1} \right\rfloor$. Then, for sufficiently large m,*

$$\mathsf{PSP}_{M,k,\ell}(\mathsf{D}) \leq s \cdot m^{O(1)} \binom{m}{\ell + 2kt\tau} \binom{\lceil \frac{w}{t} \rceil + k - 1}{k}.$$

We prove the lemma at the end of this section. As $\mathsf{D} = T_1 + \cdots + T_s$, where T_i is a product term and as $\mathsf{PSP}_{M,k,\ell}$ is sub-additive, to prove the lemma it suffices to show that for all $i \in [s]$,

$$\mathsf{PSP}_{M,k,\ell}(T_i) \leq m^{O(1)} \binom{m}{\ell + 2kt\tau} \binom{\lceil \frac{w}{t} \rceil + k - 1}{k}.$$

Consider any such product term $T = \prod_{i \in [a]} Q_i^{e_i}$, where $Q_i \in \mathbb{F}[\mathbf{x}_M, \mathbf{y}_M]$, and since D is a pruned depth four circuit, $a \leq w$. Write $Q_i = Q_i' + Q_i''$, where Q_i' is the sum of all monomials of Q_i wherein the individual degree of every $\mathbf{x}$ variable is at most two and $Q_i'' = Q_i - Q_i'$.

Then,

$$T = \prod_{i\in[a]} (Q'_i + Q''_i)^{e_i} = \prod_{i\in[a]} {Q'_i}^{e_i} + Q'',$$

where Q'' is a polynomial whose every monomial has a $\mathbf{x}$ variable with degree at least three. Thus, $\mathsf{PSP}_{M,k,\ell}(Q'') = 0$ and hence from the sub-additivity of $\mathsf{PSP}_{M,k,\ell}$ we have that

$$\mathsf{PSP}_{M,k,\ell}(T) \leq \mathsf{PSP}_{M,k,\ell}\Big(\prod_{i\in[a]} {Q'_i}^{e_i}\Big).$$

Let $T' = \prod_{i\in[a]} {Q'_i}^{e_i}$. We will now upper bound $\mathsf{PSP}_{M,k,\ell}(T')$. First, we assume without loss of generality that $a = w$ since if $a < w$ then we can multiply with additional sparse polynomials all of which are 1. Next we divide the sparse polynomials into disjoint sets such that each set (except perhaps the last) has size exactly t. Then, we have that

$$T' = P_1 \cdots P_{\lceil \frac{w}{t} \rceil}, \text{ where } P_i = \prod_{j=(i-1)t+1}^{\min(it,w)} {Q'_j}^{e_j}.$$

▷ Claim 8. Let $P = {Q'_1}^{e_1} \cdots {Q'_t}^{e_t}$ be one of the polynomials P_i. For $k \geq 0$, let $P^{(k)} := \prod_{i\in[t]} {Q'_i}^{\max(e_i-k,0)}$. Then, $\partial^k_{\mathbf{x}} P \subseteq \mathbb{F}\text{-span}\{\mathbf{y}_M^{\leq\infty}\mathbf{x}_M^{\leq k(2t\tau-1)}P^{(k)}\}$.

The proof of the claim is straighforward and is given in Appendix B.1.

Proof of Lemma 7. Recall that it is enough to show the following

$$\mathsf{PSP}_{M,k,\ell}(T') \leq m^{O(1)}\binom{m}{\ell+2kt\tau}\binom{\lceil \frac{w}{t} \rceil + k - 1}{k},$$

where $T' = P_1 \cdots P_{\lceil \frac{w}{t} \rceil}$. Let $v = \lceil \frac{w}{t} \rceil$. Now,

$$\begin{aligned}
\partial^k_{\mathbf{x}} T' &\subseteq \mathbb{F}\text{-span}\left\{\partial^{k_1}_{\mathbf{x}} P_1 \cdots \partial^{k_v}_{\mathbf{x}} P_v \;\;:\; k_1 + \cdots + k_v = k\right\} \\
&\subseteq \mathbb{F}\text{-span}\left\{\mathbf{y}_M^{\leq\infty}\mathbf{x}_M^{\leq k_1(2t\tau-1)}P_1^{(k_1)} \cdots \mathbf{y}_M^{\leq\infty}\mathbf{x}_M^{\leq k_v(2t\tau-1)}P_v^{(k_v)} \;\;:\; k_1 + \cdots + k_v = k\right\} \\
&\subseteq \mathbb{F}\text{-span}\left\{\mathbf{y}_M^{\leq\infty}\mathbf{x}_M^{\leq k(2t\tau-1)}P_1^{(k_1)} \cdots P_v^{(k_v)} \;\;:\; k_1 + \cdots + k_v = k\right\},
\end{aligned}$$

where the second to last inclusion follows from Claim 8. Hence,

$$\mathbf{x}_M^{\ell}\partial^k_{\mathbf{x}} T' \subseteq \mathbb{F}\text{-span}\left\{\mathbf{y}_M^{\leq\infty}\mathbf{x}_M^{\leq \ell+k(2t\tau-1)}P_1^{(k_1)} \cdots P_v^{(k_v)} \;\;:\; k_1 + \cdots + k_v = k\right\}.$$

In other words, the space of shifted partials of T' is contained in the $\mathbb{F}$-span of polynomials of the form $Y \cdot X \cdot P_1^{(k_1)} \cdots P_v^{(k_v)}$ where Y is a monomial in $\mathbf{y}_M$ variables and X is a monomial in $\mathbf{x}_M$ variables of degree at most $\ell + k(2t\tau - 1)$. Let us analyse the effect of the operations $\sigma_{\mathbf{y}_M=1}$, $[\cdot]_m$ and $\pi_{\mathbf{x}}$ on one such polynomial. We will assume that $\deg_{\mathbf{y}}(Y) \leq m$ and X is multilinear for otherwise the polynomial will vanish after the operations are applied. Then, we have that,

$$\begin{aligned}
&\sigma_{\mathbf{y}_M=1}\left(\left[\pi_{\mathbf{x}}\left(Y \cdot X \cdot P_1^{(k_1)} \cdots P_v^{(k_v)}\right)\right]_m\right) \\
&= X \cdot \sigma_{\mathbf{y}_M=1}\left(\left[\pi_{\mathbf{x}}\left(\sigma_{\mathrm{Supp}(X)=0}\left(P_1^{(k_1)} \cdots P_v^{(k_v)}\right)\right)\right]_{m-\deg(Y)}\right).
\end{aligned}$$

Thus,

$$\sigma_{\mathbf{y}_M=1}\left(\left[\pi_{\mathbf{x}}\left(\mathbf{x}_M^{\ell}\partial_{\mathbf{x}}^{k}T'\right)\right]_m\right) \subseteq \mathbb{F}\text{-span}\Big\{X\cdot\sigma_{\mathbf{y}_M=1}\left(\left[\pi_{\mathbf{x}}\left(\sigma_{\mathrm{Supp}(X)=0}\left(P_1^{(k_1)}\cdots P_v^{(k_v)}\right)\right)\right]_i\right): X \text{ is a multilinear monomial in } \mathbf{x}_M \text{ variables}, \deg(X) \text{ is at most } \ell+k(2t\tau-1), 0\le i\le m \text{ and } k_1+\cdots+k_v=k\Big\}.$$

Once we fix i, X, and $k_1,...,k_v$, $X\cdot\sigma_{\mathbf{y}_M=1}\left(\left[\pi_{\mathbf{x}}\left(\sigma_{\mathrm{Supp}(X)=0}\left(P_1^{(k_1)}\cdots P_v^{(k_v)}\right)\right)\right]_i\right)$ is fixed. So,

$$\begin{aligned}\mathsf{PSP}_{M,k,\ell}(T') &= \dim\left\langle\sigma_{\mathbf{y}_M=1}\left(\left[\pi_{\mathbf{x}}\left(\mathbf{x}_M^{\ell}\partial_{\mathbf{x}}^{k}T'\right)\right]_m\right)\right\rangle\\ &\le (m+1)\cdot\sum_{j=0}^{\ell+k(2t\tau-1)}\binom{m}{j}\binom{v+k-1}{k}\\ &\le (m+1)\cdot(\ell+2kt\tau)\cdot\binom{m}{\ell+2kt\tau}\binom{v+k-1}{k}\\ &= m^{O(1)}\cdot\binom{m}{\ell+2kt\tau}\binom{\lceil\frac{w}{t}\rceil+k-1}{k},\end{aligned}$$

where the second last inequality follows from Claim 9 (proved in Appendix B.1). ◀

▷ Claim 9. Let ℓ, k, t and τ be as defined earlier. Then, $\ell+2kt\tau<\frac{m}{2}$.

3.2 Pruning a depth four circuit

As mentioned before, we will prune the circuit $\mathtt{C}$ computing f_n in two steps - first we will restrict the bottom support of $\mathtt{C}$ and then we will get rid of all heavy gates in it.

3.2.1 Step 1 - Restricting the bottom support of $\mathtt{C}$

If the number of monomials in $\mathtt{C}$ is more than $\left\lfloor\frac{m^2 d_{\mathbf{x}}}{(\ln m)^5}\right\rfloor$, there is nothing to prove. Otherwise, we show that we can get rid of all monomials with support more than $\tau=\lfloor 20\ln m\rfloor$.

▶ **Lemma 10.** *Let the number of monomials in $\mathtt{C}$ be at most $\left\lfloor\frac{m^2 d_{\mathbf{x}}}{(\ln m)^5}\right\rfloor$. Then, for sufficiently large m, there exists $M_1\subseteq[3m], |M_1|=m$ such that all monomials in $\mathtt{C}_1$ obtained from $\mathtt{C}$ by setting variables $\mathbf{x}_{M_1}$ and $\mathbf{y}_{M_1}$ to 0 have support at most τ.*

Proof. We first present a greedy procedure to remove all monomials with support more than τ and then argue that it sets m variables each from $\mathbf{x}$ and $\mathbf{y}$ to 0. In each iteration the procedure picks a pair of variables that appears in a large number of monomials with support more than τ and set them to 0.

Procedure 1 Restriction procedure.

1. $M_1\leftarrow\emptyset, \mathtt{C}_1\leftarrow\mathtt{C}, H :=$ set of all monomials of $\mathtt{C}_1$ with support more than τ.
2. For $j\in[3m]$, $e(j) :=$ number of monomials in H containing x_j or y_j.
3. **while** $H\neq\emptyset$ **do**
4. Pick $j'\in[3m]\backslash M_1$ such that $e(j')\ge e(j)$ for all $j\in[3m]$. Set $x_{j'}=0$ and $y_{j'}=0$. Update $M_1\leftarrow M_1\cup\{j'\}$, $\mathtt{C}_1\leftarrow$ circuit obtained from $\mathtt{C}_1$ by setting $x_{j'}$ and $y_{j'}$ to 0, $H\leftarrow$ set of all monomials of $\mathtt{C}_1$ with support more than τ, and $e(j)\leftarrow$ number of monomials in H containing x_j or y_j.
5. **end while**

It is clear that the bottom support of $\mathtt{C}_1$ obtained after the termination of the procedure is at most τ. Also, since we are only setting variables to 0, it trivially follows that the procedure does not increase the number of gates nor does it increase the fan-in of any gate in the circuit. Claim 11 (proved in Appendix B.2) implies that the procedure terminates in at most m iterations. If it terminates before m iterations, we arbitrarily add an appropriate number of $j \in [3m]$ to M_1 so that $|M_1| = m$ and set x_j and y_j to 0 for all such j. ◀

▷ Claim 11. Procedure 1 terminates in at most m iterations.

▶ Remark. Procedure 1 looks similar to an approximation algorithm for the Set Cover problem [102] (Section 1.2, page-6). This is because the problem of removing monomials with support more than τ can be formulated as an instance of Set Cover with the universe being all such monomials and with a set corresponding to each $j \in [3m]$. For a $j \in [3m]$, the corresponding set will contain all monomials with support more than τ in which at least one of x_j and y_j appears.

3.2.2 Step 2 - Pruning the heavy gates from $\mathtt{C}_1$

A sparse polynomial Q in $\mathtt{C}_1$ is said to be *light* if its fan-in is at most $\frac{m}{(\ln m)^2}$, i.e., at most $\frac{m}{(\ln m)^2}$ many non-zero monomials are present in Q. If the sum of fan-ins of all the light sparse polynomials is $\Omega\left(\frac{m^2 d_{\mathbf{x}}}{(\ln m)^5}\right)$ then there is nothing to prove. So assume that the sum of fan-ins of all the light sparse polynomials is $O\left(\frac{m^2 d_{\mathbf{x}}}{(\ln m)^5}\right)$. Recall that a product term is called heavy if it has at least $w = \left\lfloor \frac{m d_{\mathbf{x}}}{\lambda_0 \cdot (\ln m)^3} \right\rfloor$ many distinct sparse polynomials connected to it (where λ_0 is a large enough constant fixed in Appendix C). Observe that one of the following cases is true:

1. There is a heavy gate in $\mathtt{C}_1$, that is connected to at most $\frac{m \cdot d_{\mathbf{x}}}{2 \cdot \lambda_0 \cdot (\ln m)^3}$ light sparse polynomials.
2. Every heavy gate is connected to at least $\frac{m \cdot d_{\mathbf{x}}}{2 \cdot \lambda_0 \cdot (\ln m)^3}$ light sparse polynomials.

Case 1 clearly implies a lower bound of $\Omega\big(\frac{m^2 d_{\mathbf{x}}}{(\ln m)^5}\big)$. Else, we prove the following lemma.

▶ **Lemma 12.** *Let $\mathtt{C}_1$ be the circuit (obtained from Lemma 10) having at most m heavy gates such that every heavy gate is connected to at least $\frac{m d_{\mathbf{x}}}{2 \cdot \lambda_0 \cdot (\ln m)^3}$ light sparse polynomials, and the sum of fan-ins of all the light sparse polynomials is at most $\frac{m^2 \cdot d_{\mathbf{x}}}{160 \cdot \lambda_0 \cdot (\ln m)^5}$. Then, there exist $M_2 \subseteq [3m] \setminus M_1$, $|M_2| = m$ and $\alpha_l, \beta_l \in \mathbb{F}$ for $l \in M_2$, such that setting $x_l = \alpha_l, y_l = \beta_l$ for all $l \in M_2$ removes all heavy gates from $\mathtt{C}_1$.*

Proof. We first present the pruning procedure and then argue its correctness. For any light sparse polynomial Q_j in $\mathtt{C}_1$, let b_j and c_j be equal to the fan-in of Q_j and the number of distinct heavy gates connected to Q_j in $\mathtt{C}_1$ respectively. As Q_j is a light sparse polynomial, $b_j \leq \frac{m}{(\ln m)^2}$. In this procedure, while all the heavy gates do not disappear from $\mathtt{C}_1$, we pick a sparse polynomial whose ratio of the number of distinct heavy gates connected to it and fan-in is maximum and set that to zero by evaluating it to one of its roots in $\mathbb{F}$. This can be done as $\mathbb{F}$ is algebraically closed. The restricted support of the monomials in $\mathtt{C}_1$ ensure that at the end of this procedure, only a small fraction of the variables are set.

Procedure 2 Pruning heavy gates from $\mathtt{C}_1$.

1. Set $i = 1$ and $M_2 = \emptyset$. Let $s_1 \leq m$ be the number of heavy gates in $\mathtt{C}_1$. Choose a non-constant light sparse polynomial Q_1 from $\mathtt{C}_1$ such that the ratio $\frac{c_1}{b_1}$ is maximum and add the indices of the variables appearing in Q_1 to M_2. As $b_1 \leq \frac{m}{(\ln m)^2}$, we have $\tau \cdot b_1 \leq m$.
2. Make Q_1 equal to zero by setting at most $\tau \cdot b_1$ many variables appearing in Q_1 to field constants. By doing so, at least c_1 many heavy gates vanish from $\mathtt{C}_1$.
3. **while** $(\tau(b_1 + \cdots + b_i) \leq m)$ **do**
4. Increment i by 1.
5. Let s_i be the number of heavy gates in the current circuit $\mathtt{C}_1$ obtained after the $(i-1)$-th iteration. Clearly, $s_i \leq s_{i-1} - c_{i-1}$. If $s_i = 0$ then exit the loop.
6. Otherwise, choose a non-constant light sparse polynomial Q_i from $\mathtt{C}_1$ having the maximum value of $\frac{c_i}{b_i}$ in $\mathtt{C}_1$ and add the indices of the variables appearing in Q_i to M_2.
7. Make Q_i equal to zero by setting at most $\tau \cdot b_i$ many variables appearing in Q_i to field constants. By doing so, at least c_i many heavy gates vanish from $\mathtt{C}_1$.
8. **end while**

▷ Claim 13. Let $\overline{M}_1 = [3m] \backslash M_1$. Procedure 2 sets at most m many variables in $\mathbf{x}_{\overline{M}_1} \cup \mathbf{y}_{\overline{M}_1}$ to field constants and removes all the heavy gates from $\mathtt{C}_1$.

The above claim is proved in Appendix B.3. The claim implies that $|M_2| \leq m$. If $|M_2| < m$, add appropriate number of elements from $[3m] \setminus M_1 \cup M_2$ to M_2 arbitrarily so that $|M_2| = m$. For every $l \in M_2$, if x_l or y_l is not set to a field constant then set $x_l = 0$ or $y_l = 0$ respectively. Clearly, we end up setting exactly m variables each from $\mathbf{x}_{\overline{M}_1}$ and $\mathbf{y}_{\overline{M}_1}$. ◀

▶ Remark. Procedure 2 resembles an approximation algorithm for the Weighted Set Cover problem [99] (Section 2.1, page-16). This is no coincidence as the problem of removing heavy gates can be formulated as an instance of Weighted Set Cover with the universe being all heavy gates and with a set corresponding to every sparse polynomial Q. The set corresponding to Q contains all heavy gates connected to Q and has a cost equal to the number of monomials feeding into Q.

4 An explicit polynomial family with high measure

We now describe the family $\{f_n\}_{n \geq 1}$, whose n-th member f_n is a polynomial in variables $\mathbf{x} = \{x_1, ..., x_{3m}\}$ and $\mathbf{y} = \{y_1, ..., y_{3m}\}$, where $m \in \left[\frac{n}{2}, 2n\right]$ will be fixed later.

$$f_n := \sum_{S \subseteq [3m], |S| = m} \left(\prod_{i \in S} y_i \right) \cdot \mathsf{NW}_r(\mathbf{x}_S),$$

where NW_r is a variant of the Nisan-Wigderson design polynomial (introduced in [52]), the construction of which is described later and r is a parameter fixed in this construction. Note that $\{f_n\}_{n \geq 1}$ is in VNP. Given a monomial, in order to find its coefficient in f_n, we first check if the monomial is multilinear and of degree m in $\mathbf{y}$ variables. If it is so and S is the set of the indices of the m many $\mathbf{y}$ variables in the monomial then simply return the coefficient of the part of the monomial in $\mathbf{x}$ variables in $\mathsf{NW}_r(\mathbf{x}_S)$ – this can be done as the Nisan-Wigderson polynomial family is in VNP.

Let M_1 and M_2 be as in Section 3 and $M = [3m] \setminus (M_1 \cup M_2)$. Let f_1 be the polynomial computed by the pruned circuit $\mathtt{D}$, which is obtained from $f = f_n$ by setting the variables $\mathbf{x}_{\overline{M}}$ and $\mathbf{y}_{\overline{M}}$ to field constants as in Section 3.2. Let us now see how $\mathsf{PSP}_{M,k,\ell}(f_1)$ is related to $\dim\left\langle \pi_{\mathbf{x}}\left(\mathbf{x}_M^{\ell}\partial_{\mathbf{x}}^{k}\,\mathsf{NW}_r\right)\right\rangle$.

▶ **Lemma 14.** *Let f_1 be as defined above. Then,* $\mathsf{PSP}_{M,k,\ell}(f_1) = \dim\left\langle \pi_{\mathbf{x}}\left(\mathbf{x}_M^{\ell}\partial_{\mathbf{x}}^{k}\,\mathsf{NW}_r(\mathbf{x}_M)\right)\right\rangle$.

Proof. The proof follows easily from the following two observations:

1. The two operations in $\mathbf{y}$ variables and the three operations in $\mathbf{x}$ variables (in the definition of $\mathsf{PSP}_{M,k,\ell}$) commute. That is, we have $\sigma_{\mathbf{y}_M=1}\left(\left[\pi_{\mathbf{x}}(\mathbf{x}_M^{\ell}\partial_{\mathbf{x}}^{k} f_1)\right]_m\right) = \pi_{\mathbf{x}}\left(\mathbf{x}_M^{\ell}\partial_{\mathbf{x}}^{k}\left(\sigma_{\mathbf{y}_M=1}\left([f_1]_m\right)\right)\right)$.
2. $f_1 = \left(\prod_{i\in M} y_i\right)\cdot \mathsf{NW}_r(\mathbf{x}_M) + f'$, where $f' \in \mathbb{F}[\mathbf{x}_M, \mathbf{y}_M]$ and $\deg_{\mathbf{y}}(f') < m$.

From these observations we have that

$$\begin{aligned}
\mathsf{PSP}_{M,k,\ell}(f_1) &= \dim\left\langle \sigma_{\mathbf{y}_M=1}\left(\left[\pi_{\mathbf{x}}(\mathbf{x}_M^{\ell}\partial_{\mathbf{x}}^{k} f_1)\right]_m\right)\right\rangle \\
&= \dim\left\langle \pi_{\mathbf{x}}\left(\mathbf{x}_M^{\ell}\partial_{\mathbf{x}}^{k}\left(\sigma_{\mathbf{y}_M=1}\left(\left[\left(\prod_{i\in M} y_i\right)\cdot \mathsf{NW}_r(\mathbf{x}_M) + f'\right]_m\right)\right)\right)\right\rangle \\
&= \dim\left\langle \pi_{\mathbf{x}}\left(\mathbf{x}_M^{\ell}\partial_{\mathbf{x}}^{k}\,\mathsf{NW}_r(\mathbf{x}_M)\right)\right\rangle.
\end{aligned}$$

The last equality follows from the fact that $\left(\sigma_{\mathbf{y}_M=1}\left(\left[\left(\prod_{i\in M} y_i\right)\cdot \mathsf{NW}_r(\mathbf{x}_M) + f'\right]_m\right)\right) = \mathsf{NW}_r(\mathbf{x}_M)$. ◀

Construction of NW_r. Let $d_{\mathbf{x}} = \left\lfloor \frac{\sqrt{n}}{\ln n}\right\rfloor$. Pick an α such that $d_{\mathbf{x}}\left\lceil d_{\mathbf{x}}^{1+\alpha}\right\rceil \le n \le 2d_{\mathbf{x}}\left\lceil d_{\mathbf{x}}^{1+\alpha}\right\rceil$; this forces α to be $\Theta(\frac{\ln\ln n}{\ln n})$. Let q be a prime number between $\left\lceil d_{\mathbf{x}}^{1+\alpha}\right\rceil$ and $2\left\lceil d_{\mathbf{x}}^{1+\alpha}\right\rceil$ – such a prime exists [26] – and let $m = d_{\mathbf{x}}q$. Thus, $d_{\mathbf{x}}\left\lceil d_{\mathbf{x}}^{1+\alpha}\right\rceil \le m \le 2d_{\mathbf{x}}\left\lceil d_{\mathbf{x}}^{1+\alpha}\right\rceil$ and hence $\frac{n}{2} \le m \le 2n$; moreover, it can be easily verified that $d_{\mathbf{x}} \in \left[\frac{\sqrt{m}}{2\sqrt{2}\cdot\ln m}, \frac{2\sqrt{2}\cdot\sqrt{m}}{\ln m}\right]$; both being as required in Section 3. Also notice that this means $q = \Theta(\sqrt{n}\ln n)$. Let $\beta = \frac{1}{\ln m}$ and $r = \left\lfloor \frac{\alpha+\beta}{2(1+\alpha)} d_{\mathbf{x}}\right\rfloor - 1$, $\mathbf{u} = (u_{1,1}, ..., u_{1,q}, ..., u_{d_{\mathbf{x}},1}, ..., u_{d_{\mathbf{x}},q})$ and define

$$\mathsf{NW}_r(\mathbf{u}) := \sum_{h(z)\in\mathbb{F}_q[z],\ \deg(h)\le r} u_{1,h(1)}\cdots u_{d_{\mathbf{x}},h(d_{\mathbf{x}})}.$$

A lower bound on $\dim\left\langle \pi_{\mathbf{x}}\left(\mathbf{x}_M^{\ell}\partial_{\mathbf{x}}^{k}\,\mathsf{NW}_r\right)\right\rangle$ was proved in [51, 85]. Since their analysis continues to hold for our choice of parameters – which only slightly differ from the parameters in [85] – we omit the proof of the following theorem. Moreover, while they prove this lower bound over fields of characteristic zero, the same proof also works if the characteristic is greater than $q^{(r+1)\cdot\min\left\{\binom{m}{k}\binom{m}{\ell},\binom{m}{\ell-d_{\mathbf{x}}-k}\right\}}$.

▶ **Theorem 15** (Lemma 5.2 of [51], Lemma 4.1 of [85])**.**

$$\dim\left\langle \pi_{\mathbf{x}}\left(\mathbf{x}_M^{\ell}\partial_{\mathbf{x}}^{k}\,\mathsf{NW}_r(\mathbf{x}_M)\right)\right\rangle \ge \frac{1}{m^{O(1)}}\min\left\{\frac{1}{4^k}\cdot\binom{m}{\ell}\binom{m}{k}, \binom{m}{\ell+d_{\mathbf{x}}-k}\right\}.$$

Hence, from Lemma 14 and Theorem 15 we get

▶ **Lemma 16.** $\mathsf{PSP}_{M,k,\ell}(f_1) \ge \frac{1}{m^{O(1)}}\min\left\{\frac{1}{4^k}\cdot\binom{m}{\ell}\binom{m}{k}, \binom{m}{\ell+d_{\mathbf{x}}-k}\right\}$.

5 Proof of Theorem 1

As before, let $\mathtt{C}$ be a depth four circuit computing the polynomial f_n. Before proving the theorem, let us first justify the assumption that $\mathbb{F}$ is an algebraically closed field that we made in Section 3. Suppose not. Then, let $\overline{\mathbb{F}}$ be its algebraic closure. Since $\mathtt{C}$ is also a circuit over $\overline{\mathbb{F}}$ and f_n a polynomial over $\overline{\mathbb{F}}$, we can make all arguments assuming the underlying field to be $\overline{\mathbb{F}}$. Since the size of a circuit does not depend on the underlying field, the lower bound so obtained will continue to hold when we treat $\mathtt{C}$ as a circuit over $\mathbb{F}$.

First we will prove a lower bound on the number of wires of $\mathtt{C}$. If the number of monomials in $\mathtt{C}$ is $\left\lfloor \frac{m^2 d_{\mathbf{x}}}{(\ln m)^5} \right\rfloor$ then there is nothing to prove. Otherwise from Lemma 10, we can obtain a circuit $\mathtt{C}_1$ such that the support of all the monomials of $\mathtt{C}_1$ is at most $\tau = \lfloor 20 \ln m \rfloor$, the number of gates in $\mathtt{C}_1$ is at most the number of gates in $\mathtt{C}$ and the fan-in of each gate in $\mathtt{C}_1$ is upper bounded by the fan-in of the corresponding gate in $\mathtt{C}$. Then, if $\mathtt{C}_1$ does not satisfy the hypothesis of Lemma 12, the size of $\mathtt{C}_1$ and hence the size of $\mathtt{C}$ is at least $\Omega\left(\frac{m^2 d_{\mathbf{x}}}{(\ln m)^5}\right)$. Otherwise, we can obtain a pruned circuit $\mathtt{D}$ such that the top fan-in and the bottom support of $\mathtt{D}$ are upper bounded by the top fan-in and bottom support of $\mathtt{C}_1$ and so proving a lower bound on the top fan-in of $\mathtt{D}$ would suffice.

As $\mathtt{D}$ computes f_1 (defined in Section 4), $\mathsf{PSP}_{M,k,\ell}(\mathtt{D}) = \mathsf{PSP}_{M,k,\ell}(f_1)$. Lemma 7 and 16 imply

$$s \geq \frac{\frac{1}{m^{O(1)}} \min\left\{ \frac{1}{4^k} \cdot \binom{m}{\ell}\binom{m}{k}, \binom{m}{\ell + d_{\mathbf{x}} - k} \right\}}{m^{O(1)} \cdot \binom{m}{\ell + 2kt\tau} \binom{\lceil \frac{w}{t} \rceil + k - 1}{k}}, \tag{1}$$

and the required lower bound follows from the next claim, which is proved in Appendix C.

▷ Claim 17. The top fan-in s of $\mathtt{D}$ is $\omega\left(\frac{m^2 d_{\mathbf{x}}}{(\ln m)^5}\right)$.

Now let us prove the lower bound on the number of gates. Notice that if the circuit $\mathtt{C}$ computing f has a heavy gate as defined in Section 3 then we are done. So assume that it does not have any heavy gates. Now, if the number of monomials in $\mathtt{C}$ is $\left\lfloor \frac{m^2 d_{\mathbf{x}}}{(\ln m)^5} \right\rfloor$ then there is nothing to prove. Otherwise from Lemma 10, we can obtain a circuit $\mathtt{C}_1$ such that the support of all the monomials of $\mathtt{C}_1$ is at most $\tau = \lfloor 20 \ln m \rfloor$, the number of gates in $\mathtt{C}_1$ is at most the number of gates in $\mathtt{C}$ and the fan-in of each gate in $\mathtt{C}_1$ is upper bounded by the fan-in of the corresponding gate in $\mathtt{C}$. Obtain a circuit $\mathtt{D}$ from $\mathtt{C}_1$ by picking a set $M_2 \subseteq [3m] \setminus M_1$ (where M_1 is as in Lemma 10), $|M_2| = m$ arbitrarily and setting variables in $\mathbf{x}_{M_2}$ and $\mathbf{y}_{M_2}$ to 0 (notice that the top fan-in and bottom support of $\mathtt{D}$ are upper bounded by the top fan-in and bottom support of $\mathtt{C}_1$). Now $\mathtt{D}$ computes f_1 (where f_1 is as defined in Section 4) and just as it was done in the preceding paragraph, we can show that the top fan-in of $\mathtt{D}$ is $\omega\left(\frac{m^2 d_{\mathbf{x}}}{(\ln m)^5}\right)$. However, we only get an $\Omega\left(\frac{m d_{\mathbf{x}}}{(\ln m)^3}\right)$ lower bound on the number of gates since the definition of a heavy gate is the bottleneck.

6 Conclusion

We conclude by stating a few questions/problems, some of which may not be very hard to answer/solve.

1. **Improving the depth four lower bound.** An affirmative answer to any of the following questions will strengthen or improve the lower bound shown in this work.
   - Can we prove an $\widetilde{\Omega}(n^{2.5})$ lower bound on the number of gates of depth four circuits? The almost cubic lower bound in [53] is on the number of gates of depth three circuits.

- Can we prove an $\widetilde{\Omega}(n^{2.5})$ lower bound for depth four circuits computing $\mathsf{IMM}_{2,n}$[8]? The same question may also be asked with regard to the $\widetilde{\Omega}(n^3)$ lower bound for depth three circuits.
- Can we prove an $\widetilde{\Omega}(n^3)$ lower bound for depth four circuits? The loss of a $\sqrt{n}$ factor (in Theorem 1) in comparison to the almost cubic lower bound for depth three circuits [53] is due to the use of the projected shifted partials measure in place of the shifted partials measure.

2. **A lower bound for depth five circuits.** As mentioned in Appendix A, an $\Omega(n^{1.8+\epsilon})$ lower bound on the number of gates of a depth five circuit computing $\mathsf{IMM}_{2,n}$ implies a super-cubic lower bound for depth three circuits. It is also interesting to note that the hard polynomial used in [12, 103] to prove an $\widetilde{\Omega}(n^3)$ lower bound for depth three circuits is computable by a poly(n)-size depth five circuit. As a natural next step, we pose the following problem:
   - Prove a super-quadratic lower bound for depth five circuits.
3. **Super-linear lower bound for constant depth circuits computing $\mathsf{IMM}_{2,n}$.** Recall that lower bounds of $\Omega(\Delta n^{1+\frac{1}{\Delta}})$ and roughly $\Omega(n^{1+\frac{1}{\Delta}})$ are shown for depth-Δ circuits in [86] and [73] respectively. We have argued in Appendix A that the same lower bound for depth-Δ circuits computing $\mathsf{IMM}_{2,n}$ would give a super-polynomial lower bound for constant depth circuits. It is thus natural to ask:
   - Can we prove a super-linear lower bound for constant depth circuits computing $\mathsf{IMM}_{2,n}$?
4. **Hardness magnification for commutative circuits.** It was shown in [18] that a sufficiently large super-linear lower bound for non-commutative circuits implies an arbitrarily large polynomial lower bound for general non-commutative circuits. It would be highly interesting to show a similar hardness magnification result for commutative circuits.

References

1 Manindra Agrawal, Eric Allender, and Samir Datta. On TC^0, AC^0, and Arithmetic Circuits. *J. Comput. Syst. Sci.*, 60(2):395–421, 2000. Conference version appeared in the proceedings of CCC 1997.

2 Manindra Agrawal, Chandan Saha, Ramprasad Saptharishi, and Nitin Saxena. Jacobian Hits Circuits: Hitting Sets, Lower Bounds for Depth-D Occur-k Formulas and Depth-3 Transcendence Degree-k Circuits. *SIAM J. Comput.*, 45(4):1533–1562, 2016. Conference version appeared in the proceedings of STOC 2012.

3 Manindra Agrawal and V. Vinay. Arithmetic Circuits: A Chasm at Depth Four. In *49th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2008, October 25-28, 2008, Philadelphia, PA, USA*, pages 67–75. IEEE Computer Society, 2008.

4 Miklós Ajtai. Σ^1_1-Formulae on finite structures. *Annals of Pure and Applied Logic*, 24(1):1–48, 1983.

5 Boris Alexeev, Michael A. Forbes, and Jacob Tsimerman. Tensor rank: Some lower and upper bounds. In *Proceedings of the 26th Annual IEEE Conference on Computational Complexity, CCC 2011, San Jose, California, USA, June 8-10, 2011*, pages 283–291. IEEE Computer Society, 2011.

6 Eric Allender and Michal Koucký. Amplifying lower bounds by means of self-reducibility. *J. ACM*, 57(3):14:1–14:36, 2010. Conference version appeared in the proceedings of CCC 2008.

7 Noga Alon and Ravi B. Boppana. The monotone circuit complexity of boolean functions. *Combinatorica*, 7(1):1–22, 1987.

[8] The reason why lower bounds for $\mathsf{IMM}_{2,n}$ are interesting is mentioned in the discussion on hardness magnification in Appendix A.

8 Noga Alon, Mrinal Kumar, and Ben Lee Volk. Unbalancing Sets and an Almost Quadratic Lower Bound for Syntactically Multilinear Arithmetic Circuits. In Rocco A. Servedio, editor, *33rd Computational Complexity Conference, CCC 2018, June 22-24, 2018, San Diego, CA, USA*, volume 102 of *LIPIcs*, pages 11:1–11:16. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018.

9 Matthew Anderson, Michael A. Forbes, Ramprasad Saptharishi, Amir Shpilka, and Ben Lee Volk. Identity Testing and Lower Bounds for Read-k Oblivious Algebraic Branching Programs. *TOCT*, 10(1):3:1–3:30, 2018. Conference version appeared in the proceedings of CCC 2016.

10 A. E. Andreev. On a method for obtaining more than quadratic effective lower bounds for thecomplexity of π-schemes. *Moscow Univ. Math. Bull.*, 42:63–66, 1987.

11 Vikraman Arvind and Srikanth Srinivasan. On the hardness of the noncommutative determinant. *Computational Complexity*, 27(1):1–29, 2018. Conference version appeared in the proceedings of STOC 2010.

12 Nikhil Balaji, Nutan Limaye, and Srikanth Srinivasan. An almost cubic lower bound for $\Sigma\Pi\Sigma$ circuits computing a polynomial in VP. *Electronic Colloquium on Computational Complexity (ECCC)*, 23:143, 2016. URL: `http://eccc.hpi-web.de/report/2016/143`.

13 Walter Baur and Volker Strassen. The Complexity of Partial Derivatives. *Theor. Comput. Sci.*, 22:317–330, 1983.

14 Norbert Blum. A Boolean Function Requiring 3n Network Size. *Theor. Comput. Sci.*, 28:337–345, 1984. `doi:10.1016/0304-3975(83)90029-4`.

15 Allan Borodin, Alexander A. Razborov, and Roman Smolensky. On Lower Bounds for Read-K-Times Branching Programs. *Computational Complexity*, 3:1–18, 1993.

16 Peter Bürgisser. *Completeness and Reduction in Algebraic Complexity Theory*, volume 7 of *Algorithms and computation in mathematics*. Springer, 2000.

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

18 Marco L. Carmosino, Russell Impagliazzo, Shachar Lovett, and Ivan Mihajlin. Hardness Amplification for Non-Commutative Arithmetic Circuits. In Rocco A. Servedio, editor, *33rd Computational Complexity Conference, CCC 2018, June 22-24, 2018, San Diego, CA, USA*, volume 102 of *LIPIcs*, pages 12:1–12:16. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018.

19 Prerona Chatterjee, Mrinal Kumar, Adrian She, and Ben Lee Volk. A Quadratic Lower Bound for Algebraic Branching Programs. *Electronic Colloquium on Computational Complexity (ECCC)*, page 170, 2019. URL: `https://eccc.weizmann.ac.il/report/2019/170`.

20 Lijie Chen and Roei Tell. Bootstrapping results for threshold circuits "just beyond" known lower bounds. In Moses Charikar and Edith Cohen, editors, *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019*, pages 34–41. ACM, 2019.

21 Xi Chen, Neeraj Kayal, and Avi Wigderson. Partial Derivatives in Arithmetic Complexity and Beyond. *Foundations and Trends in Theoretical Computer Science*, 6(1-2):1–138, 2011.

22 Suryajith Chillara, Christian Engels, Nutan Limaye, and Srikanth Srinivasan. A Near-Optimal Depth-Hierarchy Theorem for Small-Depth Multilinear Circuits. In Mikkel Thorup, editor, *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 934–945. IEEE Computer Society, 2018.

23 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. Conference version appeared in the proceedings of STOC 2001.

24 Danny Dolev, Cynthia Dwork, Nicholas Pippenger, and Avi Wigderson. Superconcentrators, Generalizers and Generalized Connectors with Limited Depth (Preliminary Version). In David S. Johnson, Ronald Fagin, Michael L. Fredman, David Harel, Richard M. Karp, Nancy A. Lynch, Christos H. Papadimitriou, Ronald L. Rivest, Walter L. Ruzzo, and Joel I. Seiferas, editors, *Proceedings of the 15th Annual ACM Symposium on Theory of Computing, 25-27 April, 1983, Boston, Massachusetts, USA*, pages 42–51. ACM, 1983.

25 Zeev Dvir, Guillaume Malod, Sylvain Perifel, and Amir Yehudayoff. Separating multilinear branching programs and formulas. In Howard J. Karloff and Toniann Pitassi, editors, *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, New York, NY, USA, May 19 - 22, 2012*, pages 615–624. ACM, 2012.

26 Paul Erdős. Beweis eines Satzes von Tschebyschef. *Acta Litt. Sci. Szeged*, 5:194–198, January 1932.

27 Magnus Gausdal Find, Alexander Golovnev, Edward A. Hirsch, and Alexander S. Kulikov. A Better-Than-3n Lower Bound for the Circuit Complexity of an Explicit Function. In Irit Dinur, editor, *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 89–98. IEEE Computer Society, 2016.

28 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. Conference version appeared in the proceedings of STOC 2014.

29 Merrick L. Furst, James B. Saxe, and Michael Sipser. Parity, Circuits, and the Polynomial-Time Hierarchy. *Mathematical Systems Theory*, 17(1):13–27, 1984. Conference version appeared in the proceedings of FOCS 1981.

30 Michelangelo Grigni and Michael Sipser. Monotone Separation of Logarithmic Space from Logarithmic Depth. *J. Comput. Syst. Sci.*, 50(3):433–437, 1995. Conference version appeared in the proceedings of CCC 1991.

31 Dima Grigoriev and Marek Karpinski. An Exponential Lower Bound for Depth 3 Arithmetic Circuits. In *Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing, Dallas, Texas, USA, May 23-26, 1998*, pages 577–582, 1998.

32 Dima Grigoriev and Alexander A. Razborov. Exponential Lower Bounds for Depth 3 Arithmetic Circuits in Algebras of Functions over Finite Fields. *Appl. Algebra Eng. Commun. Comput.*, 10(6):465–487, 2000. Conference version appeared in the proceedings of FOCS 1998.

33 Ankit Gupta, Pritish Kamath, Neeraj Kayal, and Ramprasad Saptharishi. Approaching the Chasm at Depth Four. *J. ACM*, 61(6):33:1–33:16, 2014. Conference version appeared in the proceedings of CCC 2013.

34 Ankit Gupta, Pritish Kamath, Neeraj Kayal, and Ramprasad Saptharishi. Arithmetic Circuits: A Chasm at Depth 3. *SIAM J. Comput.*, 45(3):1064–1079, 2016. Conference version appeared in the proceedings of FOCS 2013.

35 Johan Håstad. Almost Optimal Lower Bounds for Small Depth Circuits. In *Proceedings of the 18th Annual ACM Symposium on Theory of Computing, May 28-30, 1986, Berkeley, California, USA*, pages 6–20, 1986.

36 Johan Håstad. The Shrinkage Exponent of de Morgan Formulas is 2. *SIAM J. Comput.*, 27(1):48–64, 1998. Conference version appeared in the proceedings of FOCS 1993.

37 William Hesse, Eric Allender, and David A. Mix Barrington. Uniform constant-depth threshold circuits for division and iterated multiplication. *J. Comput. Syst. Sci.*, 65(4):695–716, 2002. Conference version appeared in the proceedings of CCC 2001.

38 Pavel Hrubes, Avi Wigderson, and Amir Yehudayoff. Non-commutative circuits and the sum-of-squares problem. In Leonard J. Schulman, editor, *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC 2010, Cambridge, Massachusetts, USA, 5-8 June 2010*, pages 667–676. ACM, 2010.

39 Pavel Hrubes and Amir Yehudayoff. Arithmetic complexity in ring extensions. *Theory of Computing*, 7(1):119–129, 2011.

40 Pavel Hrubes and Amir Yehudayoff. On Isoperimetric Profiles and Computational Complexity. In Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi, editors, *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, volume 55 of *LIPIcs*, pages 89:1–89:12, 2016.

41 Russell Impagliazzo, Ramamohan Paturi, and Michael E. Saks. Size-Depth Tradeoffs for Threshold Circuits. *SIAM J. Comput.*, 26(3):693–707, 1997. Conference version appeared in the proceedings of STOC 1993.

42 Kazuo Iwama and Hiroki Morizumi. An Explicit Lower Bound of 5n - o(n) for Boolean Circuits. In Krzysztof Diks and Wojciech Rytter, editors, *Mathematical Foundations of Computer Science 2002, 27th International Symposium, MFCS 2002, Warsaw, Poland, August 26-30, 2002, Proceedings*, volume 2420 of *Lecture Notes in Computer Science*, pages 353–364. Springer, 2002.

43 Mark Jerrum and Marc Snir. Some Exact Complexity Results for Straight-Line Computations over Semirings. *J. ACM*, 29(3):874–897, 1982.

44 K. Kalorkoti. A Lower Bound for the Formula Size of Rational Functions. *SIAM J. Comput.*, 14(3):678–687, 1985.

45 Daniel M. Kane and Ryan Williams. Super-linear gate and super-quadratic wire lower bounds for depth-two and depth-three threshold circuits. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 633–643, 2016.

46 Mauricio Karchmer and Avi Wigderson. Monotone Circuits for Connectivity Require Super-Logarithmic Depth. *SIAM J. Discrete Math.*, 3(2):255–265, 1990. Conference version appeared in the proceedings of STOC 1988.

47 Mauricio Karchmer and Avi Wigderson. On Span Programs. In *Proceedings of the Eigth Annual Structure in Complexity Theory Conference, San Diego, CA, USA, May 18-21, 1993*, pages 102–111. IEEE Computer Society, 1993.

48 Neeraj Kayal. An exponential lower bound for the sum of powers of bounded degree polynomials. *Electronic Colloquium on Computational Complexity (ECCC)*, 19:81, 2012. URL: `http://eccc.hpi-web.de/report/2012/081`.

49 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. Conference version appeared in the proceedings of FOCS 2014.

50 Neeraj Kayal and Chandan Saha. Lower Bounds for Sums of Products of Low arity Polynomials. *Electronic Colloquium on Computational Complexity (ECCC)*, 22:73, 2015. URL: `http://eccc.hpi-web.de/report/2015/073`.

51 Neeraj Kayal and Chandan Saha. Lower Bounds for Depth-Three Arithmetic Circuits with small bottom fanin. *Computational Complexity*, 25(2):419–454, 2016. Conference version appeared in the proceedings of CCC 2015.

52 Neeraj Kayal, Chandan Saha, and Ramprasad Saptharishi. A super-polynomial lower bound for regular arithmetic formulas. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 146–153, 2014.

53 Neeraj Kayal, Chandan Saha, and Sébastien Tavenas. An Almost Cubic Lower Bound for Depth Three Arithmetic Circuits. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, pages 33:1–33:15, 2016.

54 Pascal Koiran. Arithmetic circuits: The chasm at depth four gets wider. *Theor. Comput. Sci.*, 448:56–65, 2012.

55 Alexander S. Kulikov, Olga Melanich, and Ivan Mihajlin. A 5n - o(n) Lower Bound on the Circuit Size over U 2 of a Linear Boolean Function. In S. Barry Cooper, Anuj Dawar, and Benedikt Löwe, editors, *How the World Computes - Turing Centenary Conference and 8th Conference on Computability in Europe, CiE 2012, Cambridge, UK, June 18-23, 2012. Proceedings*, volume 7318 of *Lecture Notes in Computer Science*, pages 432–439. Springer, 2012.

56 Mrinal Kumar and Shubhangi Saraf. Superpolynomial lower bounds for general homogeneous depth 4 arithmetic circuits. In *Automata, Languages, and Programming - 41st International Colloquium, ICALP 2014, Copenhagen, Denmark, July 8-11, 2014, Proceedings, Part I*, pages 751–762, 2014.

57 Mrinal Kumar and Shubhangi Saraf. Sums of Products of Polynomials in Few Variables: Lower Bounds and Polynomial Identity Testing. In *31st Conference on Computational Complexity, CCC 2016, May 29 to June 1, 2016, Tokyo, Japan*, pages 35:1–35:29, 2016.

58 Mrinal Kumar and Shubhangi Saraf. On the Power of Homogeneous Depth 4 Arithmetic Circuits. *SIAM J. Comput.*, 46(1):336–387, 2017. Conference version appeared in the proceedings of FOCS 2014.

59 Mrinal Kumar and Ben Lee Volk. Lower Bounds for Matrix Factorization. *Electronic Colloquium on Computational Complexity (ECCC)*, 26:47, 2019.

60 Oded Lachish and Ran Raz. Explicit lower bound of *4.5n - o(n)* for boolean circuits. In Jeffrey Scott Vitter, Paul G. Spirakis, and Mihalis Yannakakis, editors, *Proceedings on 33rd Annual ACM Symposium on Theory of Computing, July 6-8, 2001, Heraklion, Crete, Greece*, pages 399–408. ACM, 2001.

61 Shachar Lovett. Computing polynomials with few multiplications. *Theory of Computing*, 7(1):185–188, 2011.

62 Jacques Morgenstern. Note on a Lower Bound on the Linear Complexity of the Fast Fourier Transform. *J. ACM*, 20(2):305–306, 1973.

63 Cody Murray and R. Ryan Williams. Circuit lower bounds for nondeterministic quasi-polytime: an easy witness lemma for NP and NQP. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, pages 890–901, 2018.

64 E. I. Nechiporuk. On a Boolean function. *Doklady of the Academy of Sciences of the USSR*, 164(4):765–766, 1966.

65 Noam Nisan. Lower Bounds for Non-Commutative Computation (Extended Abstract). In Cris Koutsougeras and Jeffrey Scott Vitter, editors, *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing, May 5-8, 1991, New Orleans, Louisiana, USA*, pages 410–418. ACM, 1991.

66 Noam Nisan and Avi Wigderson. Lower Bounds on Arithmetic Circuits Via Partial Derivatives. *Computational Complexity*, 6(3):217–234, 1997. Conference version appeared in the proceedings of FOCS 1995.

67 Igor Carboni Oliveira and Rahul Santhanam. Hardness Magnification for Natural Problems. In Mikkel Thorup, editor, *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 65–76. IEEE Computer Society, 2018.

68 Pavel Pudlák. Communication in Bounded Depth Circuits. *Combinatorica*, 14(2):203–216, 1994.

69 Pavel Pudlák. A note on the use of determinant for proving lower bounds on the size of linear circuits. *Inf. Process. Lett.*, 74(5-6):197–201, 2000.

70 Ran Raz. On the Complexity of Matrix Product. *SIAM J. Comput.*, 32(5):1356–1369, 2003. Conference version appeared in the proceedings of STOC 2002.

71 Ran Raz. Separation of Multilinear Circuit and Formula Size. *Theory of Computing*, 2(6):121–135, 2006. Conference version appeared in the proceedings of FOCS 2004.

72 Ran Raz. Multi-linear formulas for permanent and determinant are of super-polynomial size. *J. ACM*, 56(2):8:1–8:17, 2009. Conference version appeared in the proceedings of STOC 2004.

73 Ran Raz. Elusive Functions and Lower Bounds for Arithmetic Circuits. *Theory of Computing*, 6(1):135–177, 2010. Conference version appeared in the proceedings of STOC 2008.

74 Ran Raz. Tensor-Rank and Lower Bounds for Arithmetic Formulas. *J. ACM*, 60(6):40:1–40:15, 2013. Conference version appeared in the proceedings of STOC 2010.

75 Ran Raz and Pierre McKenzie. Separation of the Monotone NC Hierarchy. *Combinatorica*, 19(3):403–435, 1999. Conference version appeared in the proceedings of FOCS 1997.

76 Ran Raz and Amir Shpilka. Lower Bounds for Matrix Product in Bounded Depth Circuits with Arbitrary Gates. *SIAM J. Comput.*, 32(2):488–513, 2003. Conference version appeared in the proceedings of STOC 2001.

77 Ran Raz, Amir Shpilka, and Amir Yehudayoff. A Lower Bound for the Size of Syntactically Multilinear Arithmetic Circuits. *SIAM J. Comput.*, 38(4):1624–1647, 2008. Conference version appeared in the proceedings of FOCS 2007.

78 Ran Raz and Amir Yehudayoff. Balancing Syntactically Multilinear Arithmetic Circuits. *Computational Complexity*, 17(4):515–535, 2008.

79 Ran Raz and Amir Yehudayoff. Lower Bounds and Separations for Constant Depth Multilinear Circuits. *Computational Complexity*, 18(2):171–207, 2009. Conference version appeared in the proceedings of CCC 2008.

80 A. A. Razborov. Lower bounds on monotone complexity of the logical permanent. *Mathematical notes of the Academy of Sciences of the USSR*, 37(6):485–493, June 1985.

81 Alexander A. Razborov. Lower bounds on the monotone complexity of some Boolean functions. *Soviet Mathematics Doklady*, 31:354–357, 1985.

82 John H. Reif and Stephen R. Tate. On Threshold Circuits and Polynomial Computation. *SIAM J. Comput.*, 21(5):896–908, 1992.

83 Robert Robere, Toniann Pitassi, Benjamin Rossman, and Stephen A. Cook. Exponential Lower Bounds for Monotone Span Programs. In Irit Dinur, editor, *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 406–415. IEEE Computer Society, 2016.

84 Eli Shamir and Marc Snir. Lower bounds on the number of multiplications and the number of additions in monotone computations. Technical report, IBM RC 6757, 1977.

85 Abhijat Sharma. An Improved Lower Bound for Depth Four Arithmetic Circuits. Master's thesis, Indian Institute of Science, Bangalore, India, 2017.

86 Victor Shoup and Roman Smolensky. Lower Bounds for Polynomial Evaluation and Interpolation Problems. *Computational Complexity*, 6(4):301–311, 1997. Conference version appeared in the proceedings of FOCS 1991.

87 Amir Shpilka and Avi Wigderson. Depth-3 arithmetic circuits over fields of characteristic zero. *Computational Complexity*, 10(1):1–27, 2001. Conference version appeared in the proceedings of CCC 1999.

88 Amir Shpilka and Amir Yehudayoff. Arithmetic Circuits: A survey of recent results and open questions. *Foundations and Trends in Theoretical Computer Science*, 5(3-4):207–388, 2010.

89 Srikanth Srinivasan. Strongly Exponential Separation Between Monotone VP and Monotone VNP. *Electronic Colloquium on Computational Complexity (ECCC)*, 26:32, 2019.

90 Volker Strassen. Die berechnungskomplexiät von elementarysymmetrischen funktionen und von iterpolationskoeffizienten. *Numerische Mathematik*, 20:238–251, 1973.

91 Volker Strassen. Vermeidung von divisionen. *The Journal für die Reine und Angewandte Mathematik*, 264:182–202, 1973.

92 Éva Tardos. The gap between monotone and non-monotone circuit complexity is exponential. *Combinatorica*, 8(1):141–142, 1988.

93 Sébastien Tavenas. Improved bounds for reduction to depth 4 and depth 3. *Inf. Comput.*, 240:2–11, 2015. Conference version appeared in the proceedings of MFCS 2013.

94 Leslie G. Valiant. On Non-linear Lower Bounds in Computational Complexity. In William C. Rounds, Nancy Martin, Jack W. Carlyle, and Michael A. Harrison, editors, *Proceedings of the 7th Annual ACM Symposium on Theory of Computing, May 5-7, 1975, Albuquerque, New Mexico, USA*, pages 45–53. ACM, 1975.

95 Leslie G. Valiant. Graph-Theoretic Arguments in Low-Level Complexity. In *Mathematical Foundations of Computer Science 1977, 6th Symposium, Tatranska Lomnica, Czechoslovakia, September 5-9, 1977, Proceedings*, pages 162–176, 1977.

96 Leslie G. Valiant. Completeness Classes in Algebra. In *Proceedings of the 11h Annual ACM Symposium on Theory of Computing, April 30 - May 2, 1979, Atlanta, Georgia, USA*, pages 249–261, 1979.

97 Leslie G. Valiant. Negation can be exponentially powerful. *Theor. Comput. Sci.*, 12:303–314, 1980. Conference version appeared in the proceedings of STOC 1979.

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

99 Vijay V. Vazirani. *Approximation algorithms*. Springer, 2001. URL: `http://www.springer.com/computer/theoretical+computer+science/book/978-3-540-65367-7`.

100 Joachim von zur Gathen and Gadiel Seroussi. Boolean Circuits Versus Arithmetic Circuits. *Inf. Comput.*, 91(1):142–154, 1991.

101 Ryan Williams. Nonuniform ACC Circuit Lower Bounds. *J. ACM*, 61(1):2:1–2:32, 2014. Conference version appeared in the proceedings of CCC 2011.

102 David P. Williamson and David B. Shmoys. *The Design of Approximation Algorithms*. Cambridge University Press, 2011. URL: `http://www.cambridge.org/de/knowledge/isbn/item5759340/?site_locale=de_DE`.

103 Morris Yau. Almost cubic bound for depth three circuits in VP. *Electronic Colloquium on Computational Complexity (ECCC)*, 23:187, 2016. URL: `http://eccc.hpi-web.de/report/2016/187`.

104 Amir Yehudayoff. Separating monotone VP and VNP. In Moses Charikar and Edith Cohen, editors, *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019*, pages 425–429. ACM, 2019.

A Known lower bounds

We give a brief account of known lower bounds for some of the important classes of arithmetic circuits by drawing parallels with similar results from the Boolean circuit literature. The reader may refer to the surveys [21, 88] or the book [17] for more details on arithmetic circuit lower bounds. We also state a few hardness magnification[9] or amplification results to show that proving seemingly modest lower bounds can be quite interesting and challenging even for constant depth circuits, provided the polynomials being computed have "low" complexity.

General circuits. The best known lower bound for general arithmetic circuits is $\Omega(n \log d)$, which was obtained nearly four decades ago for circuits computing the power symmetric polynomial $x_1^d + x_2^d + \ldots + x_n^d$ [13, 90][10]. Recently, [19] has shown an $\Omega(n^2)$ lower bound for "layered" algebraic branching programs (ABPs) computing the same polynomial. An $\Omega(n^2)$ lower bound for formulas computing the polynomial $\sum_{i,j\in[n]} x_i^j y_j$ was shown in [44]. The situation is similar for Boolean circuits. For circuits over the DeMorgan basis and over the full binary basis, the lower bounds $5n - o(n)$ [42, 55, 60] and $(3 + \frac{1}{86})n - o(n)$ [14, 27] respectively, are the best till date. For Boolean formulas, an $\widetilde{\Omega}(n^2)$ lower bound is known over the full binary basis [64], and an $\widetilde{\Omega}(n^3)$ lower bound is known over the DeMorgan basis [10, 36].

Monotone circuits. These are arithmetic circuits over $\mathbb{Q}$ or $\mathbb{R}$ that disallow negation. A lot more is known about this class of circuits. A near optimal $2^{\Omega(n)}$ lower bound on the monotone circuit complexity of the $n \times n$ permanent was shown in [43]. In fact, [97] showed an exponential separation between monotone and general circuits computing the perfect matching polynomial of a certain planar graph. Optimal separations are also known between monotone ABPs and monotone circuits [40] and between monotone formulas and monotone ABPs [84]. Recently, [104] gave an exponential separation between monotone VP and monotone VNP which was made stronger in [89]. One of the success stories on Boolean

[9] Borrowing terminology from [67].

[10] The bound also holds for the d-th elementary symmetric polynomial in n variables.

circuit complexity in the 80s is the exponential lower bound for monotone circuits computing the clique function [7, 81]. Building on these results, [92] showed an exponential separation between monotone and general Boolean circuits[11]. An exponential separation between monotone switching networks[12] and monotone circuits was given in [83][13]. A separation between monotone formulas and monotone switching networks follows from the work of [30]. Yet another interesting result is the separation of monotone-NC^i from monotone-NC^{i+1} for every $i \leq 1$ [46, 75].

Non-commutative and multilinear circuits. A non-commutative circuit computes a polynomial in non-commuting variables. This model has more structure than circuits over commuting variables and so one may hope that it is "easier" to prove lower bounds for non-commutative circuits[14]. The seminal work of [65] showed an exponential separation between non-commutative ABPs and non-commutative circuits. But, proving a super-polynomial lower bound for general non-commutative circuits[15] and showing a separation between non-commutative formulas and non-commutative ABPs continue to remain two important open problems. The techniques used to prove lower bounds for non-commutative circuits is closely related to that used to prove lower bounds for multilinear circuits. In a (syntactically) multilinear circuit, the sets of variables occurring in the subcircuits rooted at the children of a product gate are pairwise disjoint. It is an interesting model of computation as most natural families of polynomials, like the permanent, determinant, iterated matrix multiplication, elementary symmetric polynomials, design polynomials etc., are multilinear. Building on [77], an $\Omega(\frac{n^2}{(\log n)^2})$ lower bound for multilinear circuits has been recently shown in [8]. Prior to this, the breakthrough work of [72] culminated in an optimal separation between multilinear formulas and multilinear ABPs [25, 71, 78]. We do not know of any super-polynomial lower bound for multilinear ABPs.

Bounded coefficient circuits. In a bounded coefficient circuit over $\mathbb{C}$, we forbid any multiplication by a field element having absolute value larger than 1. An $\Omega(n \log n)$ lower bound for bounded coefficient circuits computing the Discrete Fourier Transform of a vector $(x_1, \ldots, x_n)$ was shown in [62]. Later, [70] gave an $\Omega(n \log n)$ lower bound for the same class of circuits computing the product of two $\sqrt{n} \times \sqrt{n}$ matrices.

Read-k circuits. A read-once oblivious algebraic branching program (ROABP) is a layered ABP in which every layer is indexed by a variable, and every variable indexes exactly one layer. The edges of a layer are labeled by univariate polynomials in the variable indexing the layer. In a certain sense, the ROABP model generalizes quite a few other interesting arithmetic circuit models, especially tensors. An exponential lower bound for ROABPs follows from the technique introduced in the work of [65]. A read-k oblivious ABP is defined similarly, with every variable indexing at most k layers. In [9], an $\exp(\frac{n}{k^{O(k)}})$ lower bound for read-k oblivious ABP was shown. Another related result is the $\exp(\frac{n}{k^2})$ lower bound for

[11] Prior to this, [80] showed a quasi-polynomial lower bound for monotone circuits computing the perfect matching function.
[12] We may think of monotone switching networks as the Boolean analogue of monotone ABPs.
[13] In fact, [83] gave an exponential lower bound for the more powerful model of monotone span programs that was introduced in [47].
[14] Besides, there is an interesting connection between the non-commutative determinant and the commutative permanent [11].
[15] In fact, nothing better than the $\Omega(n \log d)$ lower bound (which also holds for commutative circuits) is known. The hardness of proving a sufficiently strong super-linear lower bound for non-commutative circuits is explained in a recent work [18] (see the discussion below on hardness magnification).

depth four read-k formulas [2]. An exponential lower bound is also known for read-k Boolean branching programs [15], which is a stronger model than read-k oblivious Boolean branching programs.

Constant depth circuits

The best lower bound for constant depth circuits is a little better than that for general circuits. Shoup and Smolensky [86] showed an $\Omega(\Delta n^{1+\frac{1}{\Delta}})$ lower bound for depth-Δ circuits computing the polynomials $\{\sum_{j\in[n]} x_1^j y_j, \ldots, \sum_{j\in[n]} x_n^j y_j\}$. Using an argument similar to [86], Raz [73] showed a roughly $\Omega(n^{1+\frac{1}{\Delta}})$ lower bound for depth-Δ circuits computing polynomials of degree $\Theta(\Delta)$, i.e., the polynomials have constant degree for constant depth circuits. These bounds were essentially achieved by analyzing linear circuits[16]. Prior to these works, a barely super-linear lower bound of $n \cdot \lambda_\Delta(n)$ was known for depth-Δ linear circuits using *super-concentrators* [24,68,76,94], where $\lambda_\Delta(n)$ is a very slowly growing function[17]. Recently, [59] gave a $n^{1+\frac{1}{2\Delta}}$ lower bound for depth-Δ linear circuits computing a linear transformation that can be computed in $\exp(n^{1-\Omega(\frac{1}{d})})$ time. A similar lower bound was known before for bounded coefficient circuits – Pudlák [69] proved an $\Omega(\Delta n^{1+\frac{1}{\Delta}})$ lower bound for depth-Δ bounded coefficient circuits computing the DFT matrix. A lower bound of $\Omega(n^{1+\frac{1}{O(\Delta)}})$ was also shown in [70] for depth-Δ bounded coefficient circuits computing the product of two $\sqrt{n} \times \sqrt{n}$ matrices.

A lot better lower bounds are known for constant depth multilinear circuits. Raz and Yehudayoff [79] showed an $\exp(n^{\Omega(\frac{1}{\Delta})})$ lower bound for depth-Δ multilinear circuits computing the $n \times n$ determinant polynomial. More recently, [23] showed an $\exp(n^{\frac{1}{\Delta}})$ lower bound for depth-Δ multilinear circuits computing the product of n many 2×2 matrices. In fact, an exponential separation is known between depth-Δ and depth-$(\Delta+1)$ multilinear circuits [22], which improved upon a previous quasi-polynomial separation [79].

A circuit computing an n-variate homogeneous polynomial of $\text{poly}(n)$ degree can be homogenized with only a polynomial blow-up in size [91], but this process is not depth preserving. It is plausible that homogeneous depth-Δ circuits are weaker than general depth-Δ circuits for constant Δ. Indeed, such a statement is known to be true for $\Delta = 3$ and $\Delta = 4$. It was shown in the classical work [66] that any homogeneous depth three circuit computing the n-variate degree-d elementary symmetric polynomial $\text{ESym}_{n,d}$ has size $n^{\Omega(d)}$, although $\text{ESym}_{n,d}$ has a non-homogeneous (multilinear) depth three circuit[18] of size $O(n^2)$. A sequence of work [28,33,48,49,52,58] culminated in a $n^{\Omega(\sqrt{d})}$ lower bound for homogeneous depth four circuits computing the width-n, degree-d iterated matrix multiplication polynomial $\mathsf{IMM}_{n,d}$. On the other hand, the depth reduction result in [34,93], which built on [3,54,98], yields a non-homogeneous depth four circuit of size $n^{O(d^{\frac{1}{3}})}$ for $\mathsf{IMM}_{n,d}$.

An almost cubic lower bound for general depth three circuits was shown in [53], which improved upon the previous quadratic bound [87]. The bound in [87] is for the elementary symmetric polynomial, whereas the bound in [53] is for a variant of the Nisan-Wigderson design polynomial (which is in VNP). Subsequently, [12,103] showed near cubic lower bounds for depth three circuits computing polynomials that have $\text{poly}(n)$-size depth five circuits. Over

[16] A linear circuit has only addition gates and so it computes a linear transformation (i.e., a set of linear forms) in the input variables. If a set of linear forms is computable by a circuit of size s and depth-Δ then they are computable by a linear circuit of size $O(s)$ and depth Δ. Thus, a super-linear lower bound for linear circuits implies a super-linear lower bound for general circuits.

[17] For instance, $\lambda_4(n) = \log^* n$.

[18] Construction of this circuit is attributed to Michael Ben-Or.

fixed finite fields, an exponential lower bound is known for depth three circuits computing the determinant [31,32]. For depth three circuits with low bottom fan-in[19] (but without any homogeneity restriction), [51] proved an exponential lower bound[20]. As mentioned before, the previous best lower bound for general depth four circuits is $\widetilde{\Omega}(n^{1.5})$ [85], which is a slight improvement over the roughly $\Omega(n^{1.33})$ bound obtained by specializing the lower bound for constant depth circuits in [73,86] to depth four circuits[21]. Our work here improves these super-linear bounds to a super-quadratic lower bound for depth four circuits.

Now, coming to constant depth Boolean circuits, an exponential lower bound is known for constant depth Boolean circuits over the DeMorgan basis (i.e., AC^0 circuits). However, it appears to us that the "right" Boolean analogue of constant depth arithmetic circuits is TC^0 circuits (see [1,37,82])[22]. The exponential lower bounds for AC^0 circuits [4,29,35] and the quasi-polynomial lower bounds for ACC^0 circuits[23] [63,101] are two of the great achievements in Boolean circuit complexity, but we are yet to see these kind of bounds for TC^0 circuits. The best known lower bound for threshold circuits is the slightly super-linear $n^{1+\frac{1}{c^\Delta}}$ bound for depth-Δ TC^0 circuits computing the parity function, where $c > 1$ is a fixed constant [41]. For depth two TC^0 circuits, [45] showed an $\widetilde{\Omega}(n^{2.5})$ lower bound on the number of wires and an $\widetilde{\Omega}(n^{1.5})$ lower bound on the number of gates.

Hardness magnification

There are results in the arithmetic and Boolean circuit literature that show how to obtain strong lower bounds from seemingly weak ones. We state a few of these results below with the intent of demonstrating that sufficiently strong super-linear or super-quadratic lower bounds can be quite interesting and challenging to prove even for constant depth circuits.

It follows from [91] that a cubic form[24] has a depth three powering circuit[25] with $\Theta(s)$ gates if it has a circuit of size s. Thus, a super-linear lower bound on the number of gates of a depth three powering circuit computing an explicit cubic form implies a super-linear circuit lower bound. Stated differently, a super-linear lower bound on the (symmetric) tensor rank of an explicit (symmetric) tensor of order 3 implies a super-linear circuit lower bound[26]. In fact, Raz [74] showed that an $n^{r\cdot(1-o(1))}$ lower bound on the tensor rank of an explicit order-r tensor $T : [n]^r \to \mathbb{F}$ implies a super-polynomial formula lower bound, assuming r is a super-constant and $r \leq \frac{\log n}{\log\log n}$.

---

[19] The depth reduction in [34] yields a depth three circuit with low bottom fan-in. This is reminiscent of a result in [95], which showed that a strong exponential lower bound for depth three Boolean circuits with low bottom fan-in implies a super-linear lower bound for Boolean circuits having logarithmic depth and bounded fan-in.

[20] This result was extended to depth four circuits with low bottom fan-in in the works [50,57].

[21] For the reader's convenience, we show how the $\Omega(n^{1.33})$ bound can be derived from [73,86] in Appendix D.

[22] This is because iterated addition and multiplication of integers are in TC^0, and in the converse direction, it is known that TC^0 circuits can be simulated by constant depth arithmetic circuits using a single threshold gate [1]. A related fact (attributed to Michael Ben-Or) is that there is an $O(n^2)$ size depth three arithmetic circuit computing the n-variate degree-$\frac{n}{2}$ elementary symmetric polynomial, which is the arithmetic analogue of the majority function.

[23] also for ACC^0 circuits composed with a bottom layer of threshold gates

[24] i.e., a homogeneous degree-3 polynomial

[25] A depth three powering circuit has a top +-gate, a middle layer of powering gates, and a bottom layer of +-gates.

[26] The best known lower bound for an explicit order-3 tensor $T : [n]^3 \to \mathbb{F}$ is roughly $3n$ and for an explicit order-r tensor is roughly $2n^{\lfloor \frac{r}{2} \rfloor}$ [5].

Recently, Chen and Tell [20] showed that a super-linear lower bound for TC^0 circuits (computing certain NC^1-complete functions) which is slightly better than the lower bound in [41] would imply $\mathsf{TC}^0 \neq \mathsf{NC}^1$. Their result builds on the work of [6]. By mimicking the argument in [20] for arithmetic circuits one gets the following statement: If $\mathsf{IMM}_{2,n}$ is computable[27] by a depth-Δ_0 circuit of size n^k then it is computable by a depth-Δ circuit of size $O(\frac{\Delta}{\Delta_0} \cdot n^{1+\exp(-\frac{\Delta}{\Delta_0 k})})$. Recall that an $\Omega(\Delta n^{1+\frac{1}{\Delta}})$ lower bound is already known for depth-Δ arithmetic circuits [73, 86]. If the same lower bound is shown for depth-Δ circuits *computing* $\mathsf{IMM}_{2,n}$ then that would imply a super-polynomial lower bound for constant depth arithmetic circuits! Compare this with the best known upper bound for depth-Δ circuits computing $\mathsf{IMM}_{2,n}$ which is roughly $\exp(O(\Delta n^{\frac{1}{\Delta}}))$. Even for depth three circuits, we get the following interesting observation: An $\Omega(n^{1.8+\epsilon})$ lower bound on the number of gates of a depth five circuit computing $\mathsf{IMM}_{2,n}$, for any constant $\epsilon > 0$, implies a super-cubic lower bound for depth three circuits[28].

Hardness amplification results are also known for non-commutative circuits [18, 38]. A biquadratic polynomial f in the variables $\mathbf{x} = \{x_1, \ldots, x_n\}$ and $\mathbf{y} = \{y_1, \ldots, y_n\}$ is a homogeneous degree-4 polynomial in which every monomial is of the form $x_i x_j y_k y_l$. The bilinear complexity of a biquadratic polynomial f is the minimum r such that $f = g_1 h_1 + \ldots + g_r h_r$, where g_i, h_i are bilinear forms in $\mathbf{x}$ and $\mathbf{y}$. It was shown in [38] that Permanent requires non-commutative circuits of exponential size if there is an explicit biquadratic polynomial having bilinear complexity $\Omega(n^{1+\epsilon})$, for some constant $\epsilon > 0$. In other words, a super-cubic lower bound on the size of a homogeneous depth four (commutative) circuit computing an explicit biquadratic form implies an exponential lower bound for non-commutative circuits. In another appealing instance of hardness amplification, [18] showed that an $\Omega(n^{\frac{\omega}{2}+\epsilon})$ lower bound, where ω is the matrix multiplication exponent, for non-commutative circuits computing an explicit constant degree polynomial implies an exponential lower bound for non-commutative circuits; if the explicit polynomial has $\mathrm{poly}(n)$ degree then the lower bound is an arbitrarily large polynomial function.

B Missing proofs from Section 3

B.1 Proofs from Section 3.1

▷ Claim 8. Let $P = {Q'_1}^{e_1} \cdots {Q'_t}^{e_t}$ be one of the polynomials P_i. For $k \geq 0$, let $P^{(k)} := \prod_{i \in [t]} {Q'_i}^{\max(e_i - k, 0)}$. Then, $\partial^k_{\mathbf{x}} P \subseteq \mathbb{F}\text{-span}\{\mathbf{y}_M^{\leq \infty} \mathbf{x}_M^{\leq k(2t\tau - 1)} P^{(k)}\}$.

Proof. We prove the claim by induction on k. If $k = 0$, then $\partial^0_{\mathbf{x}_M} P = \{P\} = \{P^{(0)}\}$ and hence the claim is true. Assume that the claim is true for k. Let X be a multilinear monomial of degree $k+1$ in $\mathbf{x}$ variables. Then $X = xX'$ where X' is a multilinear monomial of degree k in $\mathbf{x}$ variables and x one of the $\mathbf{x}$ variables. From the induction hypothesis we have that,

$$\frac{\partial P}{\partial X'} = g \cdot P^{(k)}$$

where g is a polynomial in $\mathbb{F}[\mathbf{x}_M, \mathbf{y}_M]$ with $\mathbf{x}_M$ degree of g being at most $k(2t\tau - 1)$ while its $\mathbf{y}_M$ degree can be arbitrarily large.

[27] Here $\mathsf{IMM}_{2,n}$ is a collection of four polynomials corresponding to the entries of a product of n many 2×2 matrices whose entries are distinct formal variables.
[28] We attribute this observation to Ankit Garg.

Let $J := \{j \in [t] : e_j > k\}$. We have that,

$$\begin{aligned}
\frac{\partial P}{\partial X} &= \frac{\partial}{\partial x}\left(g \cdot P^{(k)}\right) \\
&= \frac{\partial}{\partial x}\left(g \cdot \prod_{j\in J} {Q'_j}^{e_j-k}\right) \\
&= \frac{\partial g}{\partial x} \cdot \prod_{j\in J} {Q'_j}^{e_j-k} + g \cdot \sum_{j\in J}(e_j-k) \cdot {Q'_j}^{e_j-k-1} \cdot \frac{\partial Q'_j}{\partial x} \cdot \prod_{i\in J\setminus\{j\}} {Q'_i}^{e_i-k} \\
&= \left(\frac{\partial g}{\partial x} \cdot \prod_{j\in J} Q'_j + g \cdot \sum_{j\in J}(e_j-k) \cdot \frac{\partial Q'_j}{\partial x} \cdot \prod_{i\in J\setminus\{j\}} Q'_i\right) \cdot \prod_{j\in J} {Q'_j}^{e_j-k-1}
\end{aligned}$$

Observe that as $\mathtt{D}$ is a pruned depth four circuit, the support of all monomials of Q'_j is upper bounded by τ and as in any monomial the individual degree of any $\mathbf{x}$ variable is at most two, $\deg_{\mathbf{x}}(Q'_j) \leq 2\tau$. Also, $|J| \leq t$ and hence

$$\deg_{\mathbf{x}}\left(\frac{\partial g}{\partial x} \cdot \prod_{j\in J} Q'_j + g \cdot \sum_{j\in J}(e_j-k) \cdot \frac{\partial Q'_j}{\partial x} \cdot \prod_{i\in J\setminus\{j\}} Q'_i\right) \leq (k+1)(2t\tau-1).$$

As $\prod_{j\in J} {Q'_j}^{e_j-k-1} = P^{(k+1)}$, the claim is true for $k+1$. ◁

▷ Claim 9. Let ℓ, k, t and τ be as defined earlier. Then, $\ell + 2kt\tau < \frac{m}{2}$.

Proof. We will show that the ratio $\frac{\frac{m}{2}-2kt\tau}{\ell} > 1$. Putting the values of k and ℓ,

$$\begin{aligned}
\frac{\frac{m}{2}-2kt\tau}{\ell} &= \frac{\frac{m}{2} - 2\left\lfloor\frac{\delta d_{\mathbf{x}}}{t}\right\rfloor t\tau}{\left\lfloor\frac{m}{m^{\delta/t}+1}\right\rfloor} \\
&\geq \left(\frac{1}{2} - \frac{2\delta d_{\mathbf{x}}\tau}{m}\right)(m^{\delta/t}+1).
\end{aligned}$$

So, we need to show that

$$\begin{aligned}
\frac{1}{\frac{1}{2}-\frac{2\delta d_{\mathbf{x}}\tau}{m}} < m^{\delta/t}+1 &\iff \frac{1}{\frac{1}{2}-\frac{2\delta d_{\mathbf{x}}\tau}{m}} - 1 < m^{\delta/t} \\
&\iff \frac{1+\frac{4\delta d_{\mathbf{x}}\tau}{m}}{1-\frac{4\delta d_{\mathbf{x}}\tau}{m}} < m^{\delta/t}.
\end{aligned}$$

For large enough m, $\frac{4\delta d_{\mathbf{x}}\tau}{m} \leq \frac{1}{2}$. Using $1+x \leq e^x$, which holds for all $x \in \mathbb{R}$, and $\frac{1}{1-x} \leq e^{2x}$, which holds for $0 \leq x \leq \frac{1}{2}$ we get:

$$\frac{1+\frac{4\delta d_{\mathbf{x}}\tau}{m}}{1-\frac{4\delta d_{\mathbf{x}}\tau}{m}} \leq e^{\frac{12\delta d_{\mathbf{x}}\tau}{m}}.$$

So showing that $e^{\frac{12\delta d_{\mathbf{x}}\tau}{m}} < m^{\delta/t}$ would suffice. Now,

$$e^{\frac{12\delta d_{\mathbf{x}}\tau}{m}} < m^{\delta/t} \iff e^{\frac{12 d_{\mathbf{x}} t\tau}{m}} < m.$$

Putting the values of $d_{\mathbf{x}}, t$ an τ, we get that $\frac{12 d_{\mathbf{x}} t\tau}{m} = \frac{12 d_{\mathbf{x}}\left\lfloor\frac{d_{\mathbf{x}}}{(\ln m)^3}\right\rfloor\lfloor 20\ln m\rfloor}{m} \leq \frac{12 d_{\mathbf{x}}^2 \cdot 20\ln m}{m(\ln m)^3} = \Theta\left(\frac{m}{m(\ln m)^2(\ln m)^2}\right) = \Theta\left(\frac{1}{(\ln m)^4}\right) = o(1)$ as $d_{\mathbf{x}} = \Theta\left(\frac{\sqrt{m}}{\ln m}\right)$. Thus $e^{\frac{12 d_{\mathbf{x}} t\tau}{m}} < m$. ◁

B.2 Proof from Section 3.2.1

▷ Claim 11. Procedure 1 terminates in at most m iterations.

Proof. Let H_i be the set H after the i-th iteration of the procedure. Since each monomial in H_i has support more than τ, for any such monomial there are at least $\frac{\tau}{2}$ distinct $j \in [3m]\backslash M_1$ such that at least one of x_j and y_j appears in it. Counting the number of times at least one of x_j and y_j appears in a monomial in H_i and summing up these counts for all $j \in [3m]\backslash M_1$, we get that

$$\sum_{j\in[3m]\setminus M_1} e(j) \geq \frac{\tau \cdot |H_i|}{2};$$

so from an averaging argument there exists a j such that

$$e(j) \geq \frac{\tau \cdot |H_i|}{6m}.$$

Hence, the size of H_{i+1} is upper bounded as

$$|H_{i+1}| \leq |H_i| \cdot \left(1 - \frac{\tau}{6m}\right).$$

So after i iterations of the procedure we get,

$$\begin{aligned}
|H_i| &\leq |H_0| \cdot \left(1 - \frac{\tau}{6m}\right)^i \\
&\leq \left\lfloor \frac{m^2 d_{\mathbf{x}}}{(\ln m)^5} \right\rfloor \cdot \left(1 - \frac{\lfloor 20 \ln m \rfloor}{6m}\right)^i \\
&\leq \frac{m^2 d_{\mathbf{x}}}{(\ln m)^5} \cdot \left(1 - \frac{(20 \ln m - 1)}{6m}\right)^i \\
&\leq \frac{m^2 d_{\mathbf{x}}}{(\ln m)^5} \cdot e^{-\frac{3i \cdot \ln m}{m}} && \text{(for sufficiently large } m\text{)} \\
&= \frac{m^2 d_{\mathbf{x}}}{(\ln m)^5} \cdot m^{-\frac{3i}{m}}.
\end{aligned}$$

For $i = m, |H_i| < 1$ (for sufficiently large m), i.e., the procedure terminates in at most m iterations. ◁

B.3 Proof from Section 3.2.2

▷ Claim 13. Let $\overline{M}_1 = [3m]\backslash M_1$. Procedure 2 sets at most m many variables in $\mathbf{x}_{\overline{M}_1} \cup \mathbf{y}_{\overline{M}_1}$ to field constants and removes all the heavy gates from $\mathtt{C}_1$.

Proof. In each iteration, we evaluate a light sparse polynomial in $\mathtt{C}_1$ to zero. This can be done as $\mathbb{F}$ is an algebraically closed field. Since the support of every monomial in $\mathtt{C}_1$ is at most τ, we end up setting at most $\frac{\tau \cdot m}{(\ln m)^2} \leq \frac{20 \cdot m}{\ln m}$ many variables to field constants in each iteration. As we can afford to set m variables, Step 2 of the procedure executes successfully. For some $i \in \mathbb{N}$, the while loop terminates in the i-th iteration in either of the following two cases:

1. All the heavy gates get eliminated after the $(i-1)$-th iteration, i.e., $s_i = 0$.
2. $\tau(b_1 + \cdots + b_i) > m$. (We show in the following subclaim that all the heavy gates are eliminated before this happens. Hence, the procedure stops only in the above case.)

▷ Subclaim 18. Let $i \in \mathbb{N}$ be such that $\tau(b_1 + \cdots + b_{i-1}) \leq m$ but $\tau(b_1 + \cdots + b_i) > m$. Then, all the heavy gates in $\mathtt{C}_1$ get eliminated in the first $(i-1)$ iterations of Procedure 2. If we assume Subclaim 18 then Claim 13 is proved. ◁

Proof of Subclaim 18: For $1 \leq j < i$, let $(Q_{j,1}, \ldots, Q_{j,r_j})$ be the available light sparse polynomials in $\mathtt{C}_1$ after the $(j-1)$-th iteration. Recall that s_j is the number of heavy gates in $\mathtt{C}_1$ after the $(j-1)$-th iteration. Suppose, $s_j \geq 1$ (otherwise, we have nothing to prove). For every $l \in [r_j]$, $b_{j,l}$ and $c_{j,l}$ refer to the fan-in of $Q_{j,l}$ and the number of distinct heavy gates connected to $Q_{j,l}$ in $\mathtt{C}_1$ respectively. We may assume that $b_{j,l} \neq 0$ for every $l \in [r_j]$. It is given that

$$b_{j,1} + \ldots + b_{j,r_j} \leq \frac{m^2 d_{\mathbf{x}}}{160 \cdot \lambda_0 \cdot (\ln m)^5}. \tag{2}$$

Since every heavy gate is connected to at least $\frac{m \cdot d_{\mathbf{x}}}{2 \cdot \lambda_0 \cdot (\ln m)^3}$ many light sparse polynomials in $\mathtt{C}_1$,

$$s_j \cdot \frac{m d_{\mathbf{x}}}{2 \cdot \lambda_0 \cdot (\ln m)^3} \leq c_{j,1} + \cdots + c_{j,r_j}.$$

As $b_{j,1}, \ldots, b_{j,r_j}$ are all non-zero, we get

$$s_j \cdot \frac{m d_{\mathbf{x}}}{2 \cdot \lambda_0 \cdot (\ln m)^3} \leq \frac{c_{j,1}}{b_{j,1}} \cdot b_{j,1} + \cdots + \frac{c_{j,r_j}}{b_{j,r_j}} \cdot b_{j,r_j}.$$

Let $u \in [r_j]$ be such that $\frac{c_{j,u}}{b_{j,u}} = \max\left\{\frac{c_{j,1}}{b_{j,1}}, \ldots, \frac{c_{j,r_j}}{b_{j,r_j}}\right\}$. Let $c_j := c_{j,u}$ and $b_j := b_{j,u}$. Then, the above equation implies

$$s_j \cdot \frac{m d_{\mathbf{x}}}{2 \cdot \lambda_0 \cdot (\ln m)^3} \leq \frac{c_j}{b_j} \cdot (b_{j,1} + \cdots + b_{j,r_j}).$$

From Equation (2), we get that for every $1 \leq j < i$,

$$s_j \cdot \frac{m d_{\mathbf{x}}}{2 \cdot \lambda_0 \cdot (\ln m)^3} \leq \frac{c_j}{b_j} \cdot \frac{m^2 d_{\mathbf{x}}}{160 \cdot \lambda_0 \cdot (\ln m)^5},$$

which implies

$$\frac{80 \cdot s_j \cdot b_j \cdot (\ln m)^2}{m} \leq c_j. \tag{3}$$

Thus, by setting the light sparse polynomial $Q_{j,u}$ to zero in the j-th iteration, we get rid of at least $\frac{80 \cdot s_j \cdot b_j \cdot (\ln m)^2}{m}$ many heavy gates from $\mathtt{C}_1$. Recall that s_{j+1} is the number of available heavy gates after the j-th iteration. Then, for every $1 \leq j < i$,

$$s_{j+1} \leq s_j - c_j \leq s_j \cdot \left(1 - \frac{80 \cdot b_j \cdot (\ln m)^2}{m}\right). \tag{4}$$

Hence, for every $1 \leq j < i$,

$$s_{j+1} \leq s_1 \cdot \prod_{l=1}^{j} \left(1 - \frac{80 \cdot b_l \cdot (\ln m)^2}{m}\right). \tag{5}$$

Also, for every $l \leq j$, we have $c_l \leq s_l$. Thus, Equation (3) implies

$$\frac{80 \cdot b_l \cdot (\ln m)^2}{m} \leq 1. \tag{6}$$

As $1-a \leq e^{-\frac{a}{2}}$ for $0 \leq a \leq 1$, Equations (5) and (6) imply

$$s_{j+1} \leq s_1 \cdot \prod_{l=1}^{j} \left(e^{-\frac{40 \cdot b_l \cdot (\ln m)^2}{m}} \right) = s_1 \cdot e^{-\frac{40 \cdot (b_1 + \cdots + b_j) \cdot (\ln m)^2}{m}}. \tag{7}$$

It is given that $\tau(b_1 + \cdots + b_i) > m$, which implies $\tau(b_1 + \cdots + b_{i-1}) > m - \tau \cdot b_i$. As b_i is the fan-in of a light sparse polynomial, $b_i \leq \frac{m}{(\ln m)^2}$ and so

$$\tau(b_1 + \cdots + b_{i-1}) > m - \frac{m\tau}{(\ln m)^2}. \tag{8}$$

On substituting $j = i-1$, $\tau = \lfloor 20 \ln m \rfloor$ and the value of $\tau(b_1 + \cdots + b_{i-1})$ from Equation (8) in Equation (7), we get

$$s_i \leq s_1 \cdot e^{-\frac{40 \cdot (\ln m)^2}{m} \cdot \left(\frac{m}{20 \ln m} - \frac{m}{(\ln m)^2} \right)} = s_1 \cdot e^{-(2 \ln m - 40)}.$$

For large enough m, $s_i \leq \frac{s_1}{m^{1.9}}$. Since $s_1 \leq m$, we get $s_i = 0$ (as it is a natural number). In other words, we get rid of all the heavy gates within $(i-1)$ iterations. ◁

C Missing proofs from Section 5

▷ Claim 17. The top fan-in s of $\mathtt{D}$ is $\omega\left(\frac{m^2 d_{\mathbf{x}}}{(\ln m)^5}\right)$.

Proof. From Equation (1), we have

$$\begin{aligned}
s &\geq \frac{\frac{1}{m^{O(1)}} \min\left\{ \frac{1}{4^k} \cdot \binom{m}{\ell} \binom{m}{k}, \binom{m}{\ell + d_{\mathbf{x}} - k} \right\}}{m^{O(1)} \cdot \binom{m}{\ell + 2kt\tau} \binom{\lceil \frac{w}{t} \rceil + k - 1}{k}} \\
&\geq \frac{1}{m^{O(1)} \binom{\lceil \frac{w}{t} \rceil + k - 1}{k}} \min\left\{ \frac{\binom{m}{k}}{4^k} \cdot \frac{\binom{m}{\ell+1}}{\binom{m}{\ell + 2kt\tau + 1}}, \frac{\binom{m}{\ell + d_{\mathbf{x}} - k}}{\binom{m}{\ell + 2kt\tau}} \right\} \\
&= \frac{1}{m^{O(1)} \binom{\lceil \frac{w}{t} \rceil + k - 1}{k}} \min\left\{ \frac{\binom{m}{k}}{4^k} \cdot \frac{(m - \ell - 2kt\tau - 1)!}{(m - \ell - 1)!} \cdot \frac{(\ell + 2kt\tau + 1)!}{(\ell + 1)!}, \right. \\
&\qquad \left. \frac{(m - \ell - 2kt\tau)!}{(m - \ell - d_{\mathbf{x}} + k)!} \cdot \frac{(\ell + 2kt\tau)!}{(\ell + d_{\mathbf{x}} - k)!} \right\} \\
&= \frac{1}{m^{O(1)} \binom{\lceil \frac{w}{t} \rceil + k - 1}{k}} \min\left\{ \frac{\binom{m}{k}}{4^k} \cdot e^{(-2kt\tau) \ln \frac{m-\ell-1}{\ell+1} \pm o(1)}, e^{(d_{\mathbf{x}} - 2kt\tau - k) \ln \frac{m-\ell}{\ell} \pm o(1)} \right\} \\
&\qquad \text{(Using Proposition 5.)} \\
&\geq \frac{1}{m^{O(1)} \binom{\lceil \frac{w}{t} \rceil + k - 1}{k}} \min\left\{ \frac{\binom{m}{k}}{4^k} \cdot \left(\frac{m}{\ell + 1} - 1 \right)^{-2kt\tau}, \left(\frac{m}{\ell} - 1 \right)^{(d_{\mathbf{x}} - 2kt\tau - k)} \right\} \\
&= \frac{1}{m^{O(1)} \binom{\lceil \frac{w}{t} \rceil + k - 1}{k}} \min\left\{ \frac{\binom{m}{k}}{4^k} \cdot \left(\frac{m}{\left\lfloor \frac{m}{m^{\delta/t} + 1} \right\rfloor + 1} - 1 \right)^{-2kt\tau}, \left(\frac{m}{\left\lfloor \frac{m}{m^{\delta/t} + 1} \right\rfloor} - 1 \right)^{(d_{\mathbf{x}} - 2kt\tau - k)} \right\} \\
&\geq \frac{1}{m^{O(1)} \binom{\lceil \frac{w}{t} \rceil + k - 1}{k}} \min\left\{ \frac{\binom{m}{k}}{4^k} \cdot \left(\frac{m}{\frac{m}{m^{\delta/t} + 1}} - 1 \right)^{-2kt\tau}, \left(\frac{m}{\frac{m}{m^{\delta/t} + 1}} - 1 \right)^{(d_{\mathbf{x}} - 2kt\tau - k)} \right\} \\
&\geq \frac{1}{m^{O(1)} \binom{\lceil \frac{w}{t} \rceil + k - 1}{k}} \min\left\{ \frac{\binom{m}{k}}{4^k} \cdot m^{-2k\delta\tau}, m^{(1 - 2\delta\tau - \frac{\delta}{t})k} \right\}
\end{aligned}$$

Since $\frac{\binom{m}{k}}{4^k} \cdot m^{-2k\delta\tau} = \frac{\binom{m}{k}}{4^k \cdot m^{(1-\frac{\delta}{t})k}} \cdot m^{(1-2\delta\tau-\frac{\delta}{t})k} \leq (\frac{em}{k})^k \cdot \frac{m^{\frac{\delta k}{t}}}{4^k m^k} \cdot m^{(1-2\delta\tau-\frac{\delta}{t})k}$. For our choice of parameters δ, k and t, $m^{\frac{\delta k}{t}} = O(1)$. Hence, $\frac{\binom{m}{k}}{4^k} \cdot m^{-2k\delta\tau} \leq m^{(1-2\delta\tau-\frac{\delta}{t})k}$ and thus,

$$
\begin{aligned}
s &\geq \frac{1}{m^{O(1)}} \cdot \frac{\binom{m}{k} \cdot m^{-2k\delta\tau}}{4^k \cdot \binom{\lceil \frac{w}{t} \rceil + k - 1}{k}} \\
&\geq \frac{1}{m^{O(1)}} \cdot \left(\frac{m \cdot k}{4e \cdot k \cdot m^{2\delta\tau} \cdot (\frac{w}{t} + k)} \right)^k && \text{(Using Proposition 4.)} \\
&\geq \frac{1}{m^{O(1)}} \cdot \left(\frac{m \cdot t}{8e \cdot m^{2\delta\tau} \cdot w} \right)^k && \text{(Since } kt \leq w = \left\lfloor \tfrac{m d_{\mathbf{x}}}{\lambda_0 \cdot (\ln m)^3} \right\rfloor .) \\
&= \frac{1}{m^{O(1)}} \cdot \left(\frac{m \cdot \left\lfloor \frac{d_{\mathbf{x}}}{(\ln m)^3} \right\rfloor}{8e \cdot m^{2\delta\tau} \cdot \left\lfloor \frac{m d_{\mathbf{x}}}{\lambda_0 \cdot (\ln m)^3} \right\rfloor} \right)^k \\
&\geq \frac{1}{m^{O(1)}} \cdot \left(\frac{m \cdot \frac{d_{\mathbf{x}}}{(\ln m)^3}}{16e \cdot m^{2\delta\tau} \cdot \frac{m d_{\mathbf{x}}}{\lambda_0 \cdot (\ln m)^3}} \right)^k \\
&\geq \frac{1}{m^{O(1)}} \cdot \left(\frac{\lambda_0}{16e \cdot m^{2 \cdot \frac{1}{(\ln m)^2} \cdot \lfloor 20 \ln m \rfloor}} \right)^{\ln m} && \text{(Since } k \geq \lfloor \ln m \rfloor .) \\
&= \frac{1}{m^{O(1)}} \cdot \left(\frac{\lambda_0}{16e \cdot e^{O(1)}} \right)^{\ln m} \\
&= \omega\left(\frac{m^2 d_{\mathbf{x}}}{(\ln m)^5} \right),
\end{aligned}
$$

if we choose λ_0 to be a large enough constant. ◁

D A brief review of the lower bounds from [86] and [73]

In this section, we present a short overview of the lower bounds for restricted depth arithmetic circuits with multiple output gates from [86] and [73] and focus mainly on depth four circuits. We would use (s, Δ)-arithmetic circuit to denote an arithmetic circuit of size-s and depth-Δ and $\mathbf{y}$ for the set of variables $\{y_1, \ldots, y_n\}$.

Lower bound from [86]. Let $n \in \mathbb{N}$ and $\Delta = O(\log n)$. Shoup and Smolensky showed that there exist n linear forms $g_1, \ldots, g_n \in \mathbb{C}[\mathbf{y}]$, such that the size of any depth-Δ normal-linear circuit[29] that computes $g_1, \ldots, g_n$ is $\Omega(\Delta n^{1+\frac{1}{\Delta}})$. The following proposition implies that the same lower bound holds for a depth-Δ arithmetic circuit, that also computes $g_1, \ldots, g_n$.

▶ **Proposition 19.** *Let $n \in \mathbb{N}$, $\mathbb{F}$ be an arbitrary field and $h_1, \ldots, h_n \in \mathbb{F}[\mathbf{y}]$ be linear forms computed by an (s, Δ)-arithmetic circuit. Then, there exists an (s, Δ)-normal-linear circuit that computes $h_1, \ldots, h_n$.*

This proposition is easy to prove; a proof of the same in given in Section 2 of [73]. We refer the reader to Section 3 of [86] for more details. In case of depth four arithmetic circuits over $\mathbb{C}$, if we substitute $\Delta = 4$ in the above mentioned result then we get a lower bound of

[29] An arithmetic circuit D over $\mathbb{F}$ is called a normal-linear circuit if all the gates in D are labelled by either variables or by $+$. Every gate in D computes a linear form in the underlying set of variables over $\mathbb{F}$.

$\Omega(n^{1.25})$, but we observe that this lower bound can be optimised roughly to $\Omega(n^{1.33})$ using the following claim. Claim 20 implies that the size of any depth four arithmetic circuit and any depth three normal-linear circuit computing the linear forms $g_1, \ldots, g_n$ given in [86] are same, which is roughly $\Omega(n^{1.33})$.

▷ Claim 20. Let $n \in \mathbb{N}$ and $h_1, \ldots, h_n \in \mathbb{F}[\mathbf{y}]$ be linear forms, computed by an $(s, 4)$-arithmetic circuit $\mathtt{C}$ over $\mathbb{F}$. Then, there exists an $(s, 3)$-normal-linear circuit over $\mathbb{F}$ that computes $h_1, \ldots, h_n$.

Proof. From Proposition 19, we obtain an $(s, 4)$-normal-linear circuit $\mathtt{D}$ over $\mathbb{F}$ that computes $h_1, \ldots, h_n$. We now argue that linearisation ensures that the fan-in of every gate in the bottom layer of $\mathtt{D}$ is exactly 1. It turns out that only those product gates survive the linearisation in the bottom layer of $\mathtt{C}$ which are connected to exactly one variable. Let v be a gate in the bottom layer of $\mathtt{C}$ with children $u_1, \ldots, u_r$. Then, there exists some $i \in [r]$ such that u_i is a variable and all other gates are labelled by field constants, in which case, we remove $u_j, j \in [r] \setminus \{i\}$ and multiply the label of the edge (v, u_i) with $\prod_{j \in [r]\setminus\{i\}} u_j$. As every gate in the bottom layer of $\mathtt{D}$ has fan-in 1, we can directly connect the input of every gate in this layer to its outputs, thereby yielding an $(s, 3)$-normal-linear circuit that computes $h_1, \ldots, h_n$. ◁

Lower bound from [73]. Let n be a prime number and $\Delta = O(\log n)$. Raz showed that there exist n explicit homogeneous polynomials of degree $\Theta(\Delta)$ in $\Theta(n)$ variables over $\mathbb{F}$, such that any depth-Δ arithmetic circuit that computes these polynomials has size $\Omega(n^{1+\frac{1}{2\cdot\Delta}})$. While the lower bound for depth-Δ arithmetic circuit given in [86] holds for n non-explicit linear forms over $\mathbb{C}$, the same lower bound also holds for n explicit homogeneous polynomials of $\Theta(n)$ degree in $\Theta(n)$ variables over $\mathbb{C}$. We first recall some definitions from [73] and then show that the lower bound for depth-Δ arithmetic circuits in [73] can be optimized slightly.

Let $n', m, t, s \in \mathbb{N}$. A polynomial mapping $f : \mathbb{F}^{n'} \to \mathbb{F}^m$ of degree t is an m tuple $(f_1, \ldots, f_m)$ of n' variate degree t polynomials over $\mathbb{F}$. The polynomial mapping f *eludes* a polynomial mapping $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$ if $Image(f) \not\subset Image(\Gamma)$. Moreover, f is said to be (s, t)-*elusive* over $\mathbb{F}$ if it eludes every polynomial mapping $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$ of degree at most t.

Let n be a prime, $\Delta = O(\log n)$, $m := n^2, \Delta' := a \cdot \Delta$, where $a \in \mathbb{N}$ is a constant, $n' := \Delta' \cdot n$ and $\mathbf{x} := \{x_{k,l} : k \in [\Delta'], l \in [n]\}$. Let $f : \mathbb{F}^{n'} \to \mathbb{F}^m$ be defined as follows:

For every $(i, j) \in [n] \times [n]$,

$$f_{(i,j)}(\mathbf{x}) := \prod_{k=1}^{\Delta'} x_{k,(i+j\cdot k) \bmod n}. \tag{9}$$

Further, for every $i \in [n]$,

$$\tilde{f}_i(\mathbf{x}, \mathbf{y}) := \sum_{j \in [n]} y_j \cdot f_{(i,j)}(\mathbf{x}). \tag{10}$$

[73] showed that for $a = 5$, any depth-Δ arithmetic circuit computing $\tilde{f}_1, \ldots, \tilde{f}_n$ requires size $\Omega(n^{1+\frac{1}{2\cdot\Delta}})$. The detailed proof is given in Section 4 of [73]. Here, we show how this lower bound can be optimized to $\Omega(n^{1+\frac{1}{\Delta}-\epsilon_{a,\Delta}})$, where $\epsilon_{a,\Delta} := \frac{2\cdot\Delta-1}{a\cdot\Delta^2}$. Note that as we increase the value of a, this lower bound gets closer to the one for depth-Δ arithmetic circuits given in [86]. The main ingredient of this improvement is the following optimization of Lemma 4.1 in [73].

▶ **Lemma 21.** *Let n be a prime, $m = n^2$, $\Delta = O(\log n)$, $\Delta' = a \cdot \Delta$, where $a \in \mathbb{N}$ is a constant and $n' = \Delta' \cdot n$. Let $\mathbb{G}$ be a field extension of $\mathbb{F}$ of size more than m and $f : \mathbb{G}^{n'} \to \mathbb{G}^m$ be the polynomial mapping defined in Equation (9)*[30]*. Then, f is (s, Δ)-elusive over $\mathbb{G}$, where $s = \lfloor n^{1+\frac{1}{\Delta}-\epsilon_{a,\Delta}} \rfloor$ and $\epsilon_{a,\Delta} = \frac{2\cdot\Delta-1}{a\cdot\Delta^2}$.*

Proof. Let $U := [n] \times [n]$, $r := \frac{1}{2}\lfloor n^{1-\frac{2}{\Delta'}} \rfloor$. For $A \subseteq U$, $f_A(\mathbf{x}) := \prod_{(i,j)\in A} f_{i,j}(\mathbf{x})$. A is said to be *retrievable* if for any $A' \subseteq U$, $f_A \neq f_{A'}$ implies $A \neq A'$. It is shown in Claim 4.2 of [73] that

$$\Pr_{A \in_R \binom{U}{r}}[A \text{ is not retrievable}] \leq \left(\frac{|A|}{n+1}\right)^{\Delta'} \cdot n^2,$$

where $A \in_R \binom{U}{r}$ means that A is a subset of U of size r chosen uniformly at random. On plugging the value of r in the above equation, we get

$$\Pr_{A \in_R \binom{U}{r}}[A \text{ is retrievable}] > \frac{1}{2}. \tag{11}$$

Let $\mathcal{L}$ be the set of degree r multilinear homogeneous polynomials of the type $g : \mathbb{G}^m \to \mathbb{G}$, such that every monomial of g corresponds to a retrievable set. Clearly, $\mathcal{L}$ is a $\mathbb{G}$-vector space. From Equation (11), we get $\dim_{\mathbb{G}}(\mathcal{L}) > \frac{1}{2}\binom{m}{r} \geq \frac{1}{2}\left(\frac{m}{r}\right)^r = \frac{1}{2}\left(2n^{1+\frac{2}{\Delta'}}\right)^r$. Fix a polynomial map $\Gamma : \mathbb{G}^s \to \mathbb{G}^m$ of degree Δ. Then, for every $g \in \mathcal{L}$, $g \circ \Gamma : \mathbb{G}^s \to \mathbb{G}$ is a polynomial of degree $r \cdot \Delta$. Let $\mathcal{K}$ be the set of all polynomials from $\mathbb{G}^s$ to $\mathbb{G}$ of degree at most $r \cdot \Delta$. Then, $\mathcal{K}$ is a $\mathbb{G}$-vector space and $\dim_{\mathbb{G}} \mathcal{K} \leq \binom{s+r\cdot\Delta}{r\cdot\Delta} \leq \left(\frac{e(s+r\cdot\Delta)}{r\cdot\Delta}\right)^{r\cdot\Delta} < \left(\frac{2es}{r}\right)^{r\cdot\Delta} = \left(12n^{\frac{1}{\Delta}+\frac{2}{\Delta'}-\epsilon_{a,\Delta}}\right)^{r\cdot\Delta}$.

On substituting the values of Δ' and $\epsilon_{a,\Delta}$ in $\dim_{\mathbb{G}} \mathcal{L}$ and $\dim_{\mathbb{G}} \mathcal{K}$, we get $\dim_{\mathbb{G}} \mathcal{K} < \dim_{\mathbb{G}} \mathcal{L}$. Now, for a fixed polynomial map $\Gamma : \mathbb{G}^s \to \mathbb{G}^m$ of degree Δ, define $\varphi_\Gamma : \mathcal{L} \to \mathcal{K}\,; g \mapsto g \circ \Gamma$. Clearly, φ_Γ is a $\mathbb{G}$-linear map and as $\dim_{\mathbb{G}} \mathcal{K} < \dim_{\mathbb{G}} \mathcal{L}$, φ_Γ is not an injective map. This means that there exists a non-zero $g_\Gamma \in \mathcal{L}$, such that $\varphi_\Gamma(g_\Gamma) = g_\Gamma \circ \Gamma = 0$. As $|\mathbb{G}| > m$, Claim 4.4 in [73] implies that $g_\Gamma \circ f : \mathbb{G}^{n'} \to \mathbb{G}$ is not the zero polynomial. Thus, for every polynomial mapping $\Gamma : \mathbb{G}^s \to \mathbb{G}^m$ of degree Δ, $Image(f) \not\subset Image(\Gamma)$. Hence, f is an (s, Δ)-elusive polynomial map over $\mathbb{G}$. ◀

The following is a corollary of Lemma 21 and Proposition 3.11 in [73].

▶ **Corollary 22.** *Let n be a prime, $\Delta = O(\log n)$ and $\Delta' = a \cdot \Delta$ for some constant $a \in \mathbb{N}$. Let $\tilde{f}_1, \ldots, \tilde{f}_n$ be $n(\Delta'+1)$ variate degree $\Delta'+1$ polynomials as defined in Equation (10). Then, any depth-Δ arithmetic circuit $\mathtt{C}$ over $\mathbb{F}$ computing $\tilde{f}_1, \ldots, \tilde{f}_n$ requires size $\Omega\left(n^{1+\frac{1}{\Delta}-\epsilon_{a,\Delta}}\right)$, where $\epsilon_{a,\Delta} = \frac{2\cdot\Delta-1}{a\cdot\Delta^2}$.*

Proof idea. In Proposition 3.11 in [73], $\tilde{f}_1, \ldots, \tilde{f}_n$ and $\mathtt{C}$ are viewed as linear polynomials in $\mathbf{y}$ variables over the function field $\mathbb{F}(\mathbf{x})$ and an arithmetic circuit over $\mathbb{F}(\mathbf{x})$ respectively. Then, using Proposition 19, $\mathtt{C}$ is converted to an (s, Δ)-normal-linear circuit over $\mathbb{F}(\mathbf{x})$, that also computes $\tilde{f}_1, \ldots, \tilde{f}_n$. After that, on invoking Lemma 21, we get the lower bound of $\Omega\left(n^{1+\frac{1}{\Delta}-\epsilon_{a,\Delta}}\right)$ on a depth-Δ arithmetic circuit computing $\tilde{f}_1, \ldots, \tilde{f}_n$. ◀

We now focus on depth four circuits. Let $s := n^{\frac{4}{3}-\epsilon_{a,3}}$, where $\epsilon_{a,3} := \frac{5}{9a}$. In the proof of Corollary 22, we use Claim 20 to obtain an $(s, 3)$-normal-linear circuit over $\mathbb{F}(\mathbf{x})$ from an $(s, 4)$-arithmetic circuit over $\mathbb{F}(\mathbf{x})$, such that both circuits compute $\tilde{f}_1, \ldots, \tilde{f}_n$. As Lemma 21 implies that the polynomial mapping f is $(s, 3)$-elusive, we get a lower bound of $\Omega(n^{\frac{4}{3}-\epsilon_{a,3}})$ for depth four circuits.

[30] f is naturally a polynomial mapping over $\mathbb{G}$ because from every $i, j \in [n]$, $f_{i,j}(\mathbf{x}) \in \mathbb{F}[\mathbf{x}] \subseteq \mathbb{G}[\mathbf{x}]$.