An improved lower bound for multi-r-ic depth four circuits as a function of the number of input variables

A THESIS

SUBMITTED FOR THE DEGREE OF

Master of Science (Engineering)

IN THE

Faculty of Engineering

BY

Sumant Hegde

Computer Science and Automation
Indian Institute of Science
Bangalore – 560 012 (INDIA)

April, 2017

To my parents

Acknowledgements

This thesis would not have been possible without the constant guidance and support from my advisor Chandan Saha. I want to thank him for agreeing to be my research advisor, showing how research is to be done, and helping me improve my research and mathematical aptitude. The interactions I had with him and his way of leading by example have implicitly educated me about numerous aspects of life, and made me a better person.

I take this opportunity to also remember all my teachers who hugely influenced my thinking early in my childhood. I am indebted to Vinayak Hegde, who sparked in me an interest for mathematics, boosted my confidence and instilled in me the sense of humor. R. S. Hegde and Mahadevi Markande appreciated and fuelled my interest in science. N. N. Hegde Kannimane and M. M. Hegde Khasapal have been my well-wishers. I would like to thank all of them.

Several people have gone an extra mile and given me a helping hand in my difficult times, at various points in my career. I want to particularly mention Shripad Bhat, Ramkishan B. K. and Nagachetan, and thank them.

Thanks to this institute with a wonderful campus for offering me the opportunity to pursue graduate program. I want to thank Arnab Bhattacharyya and Sathish Govindarajan for teaching me algorithms, Bhavana Kanukurthi and Arpita Patra for teaching me discrete mathematics, Dilip Patil for teaching algebraic geometry and Neeraj Kayal and Chandan Saha for teaching me computational complexity theory. Thanks also to the staff of CSA for all the help.

My stay here is made more memorable and enjoyable by my friends. Without mentioning all their names I express my gratitude to them. Yet, I feel compelled to mention Shivaraj Kumar and Kiran Shiragur and thank them for their valuable advice. I also want to thank my fellow lab members and classmates – Vineet, Abhijat, Nikhil, Mayank and Saravana to name just a few – for wonderful discussions and time spent together.

Last but not the least, I want to express my deepest gratitude to my family for their constant love and care. I owe a lot to my brother, I have immensely benefitted from his continual love and

support. I am extremely indebted to my parents for loving me, caring for me and encouraging me to pursue what I feel passionate about. If not for their support, I would not have enjoyed the sense of security and freedom that I have throughout the years. I dedicate this thesis to them.

Abstract

In this work we study the multi-r-ic formula model introduced by [KS15c] and improve upon the lower bound for multi-r-ic depth four circuits given in [KST16b], when viewed as a function of the number of input variables N. The improvement leads to superpolynomial lower bounds for values of r significantly higher than what is known from prior works.

A (syntactically) multi-r-ic formula is an arithmetic formula in which the formal degree with respect to every variable is at most r at every gate. The formal degree of an input gate with respect to a variable x is defined to be 1 if the gate is labelled with x and 0 if it is labelled with a field element or a different variable. The formal degree of a sum (respectively, product) gate with respect to x is defined as the maximum (respectively, sum) of the formal degrees of its children with respect to x. A multi-r-ic formula computes a polynomial with individual degree of every variable bounded by r.

Multi-r-ic formulas are a natural extension of the relatively well-studied multilinear formulas [Raz09, RY09]. In this work, we focus on multi-r-ic formulas that compute multilinear polynomials. They are interesting because they allow the formal degree of the formula to be as high as r times the number of underlying variables. This gives extra room for 'clever' cancellations of the high degree components inside the formula thereby making this type of formulas harder to analyze (as formula homogenization is not known to be doable without blowing up the size superpolynomially unless degree is very small [Raz10]). Most lower bound proofs in the literature operate under the restriction of low formal degree or multilinearity [Raz09, RY09, KSS14, KLSS]. In this light, multi-r-ic formulas computing multilinear polynomials form a reasonable intermediate model to study in order to gain some insight on how to deal with high formal degree in general formulas. Another motivation for understanding the high formal degree case better (even at depth three) comes from the depth reduction result in [GKKS14].

With the aim of making progress on multi-r-ic formula lower bound, [KST16b] gave a $(\frac{N}{d \cdot r^2})^{\Omega(\sqrt{d/r})}$ lower bound for multi-r-ic depth four formulas computing the N-variate Iterated

Abstract

Matrix Multiplication (IMM) polynomial of degree d. As a function of N, the lower bound is at most $2^{\Omega(\sqrt{N/r^3})}$ when $d = \Theta(N/r^2)$. In this thesis, our focus is on getting multi-r-ic depth four formulas with *larger* r into the arena of models that provenly admit a superpolynomial lower bound. In [KST16b], r can be at most $N^{1/3}$ for the bound to remain superpolynomial. Our result (stated below) gives a superpolynomial lower bound for multi-r-ic depth four formulas where r can be as high as $(N \cdot \log N)^{0.9}$.

> **Theorem.** Let N, d, r be positive integers such that $0.51 \cdot N \leq d \leq 0.9 \cdot N$ and $r \leq (N \cdot \log N)^{0.9}$. Then there is an explicit N-variate degree-d multilinear polynomial in VNP such that any multi-r-ic depth four circuit computing it has size $2^{\Omega\left(\sqrt{\frac{N \cdot \log N}{r}}\right)}$.

The theorem yields a better lower bound than that of [KST16b], when viewed as a function of N. Also, the bound matches the best known lower bound (as a function of N) for multilinear ($r = 1$) depth four circuits [RY09] which is $2^{\Omega(\sqrt{N \cdot \log N})}$.

The improvement is obtained by analyzing the shifted partials dimension (SPD) of an N-variate polynomial in VNP (as opposed to a VP polynomial in [KST16b]) of high degree range of $\Theta(N)$, and comparing it with the SPD of a depth four multi-r-ic circuit. In [KST16b] a variant of shifted partials, called shifted *skewed* partials, is critically used to analyze the IMM polynomial (which is in VP) and obtain a lower bound as a function of N and d (particularly for low d). We observe that SPD (without 'skew') is still effective for the Nisan-Wigderson polynomial (which is in VNP), and yields a better lower bound as a function of only N when degree d is naturally chosen to be high.

Our analysis gives a better range for r and a better lower bound in the high degree regime, not only for depth four multi-r-ic circuits but also for the weaker models: multi-r-ic depth three circuits and multi-r-ic depth four circuits with low bottom support. These (weaker) models are instrumental in gaining insight about general depth four multi-r-ic circuits, both in [KST16b] and our work.

Contents

Acknowledgements i

Abstract iii

Contents v

List of Tables vii

1 Introduction 1

1.1 Previous Works . . . 5
1.2 Our results . . . 7
1.3 Discussions . . . 8
1.4 Outline of the rest of the thesis . . . 12

2 Preliminaries 13

2.1 Notations . . . 13
2.2 Some well-known bounds . . . 13
2.3 Arithmetic circuits . . . 14
2.4 Arithmetic complexity classes . . . 15
2.5 The shifted partials dimension measure . . . 17

3 Depth three Multi-r-ic circuits 21

3.1 Model . . . 21
3.2 Upper bounding SPD of a term . . . 22
3.3 A lower bound on SPD of a hard polynomial . . . 25
3.4 Putting things together . . . 25
3.5 SPD of a depth three circuit vs. the maximum SPD . . . 27

4 Depth four multi-r-ic circuits with low bottom support **28**
4.1 Model . . . 28
4.2 Upper bounding SPD of a term . . . 29
4.3 A lower bound on SPD of a hard polynomial . . . 32
4.4 Putting things together . . . 32

5 A polynomial family with large SPD **34**
5.1 SPD and the pairwise monomial distance . . . 34
5.2 Proving Theorem 12 . . . 37
5.3 Monomials with large pairwise distance . . . 38
5.3.1 Proof of Lemma 19 . . . 41

6 Depth four multi-r-ic circuits **44**
6.1 Polynomial H . . . 45
6.2 Proof of Theorem 1 . . . 46
6.3 Proof of Lemma 23 . . . 47

7 Conclusion **49**

Bibliography **51**

List of Tables

1.1 Comparison of our results with [KST16b] 8

Chapter 1

Introduction

The role of polynomials in computer applications cannot be overstated. Think of an image-processing application for instance. In all probability it has a piece of code that takes as input a square matrix of a certain size and outputs its determinant. Viewing individual matrix entries as variables, the determinant is a polynomial expression in those variables. The piece of code is effectively evaluating the (determinant) polynomial at given input values of variables. The determinant is only an example; in fact every arithmetic/algebraic operation is equivalently a process of evaluating, or *computing*, some polynomial.

A natural question then, from an algorithmist's standpoint, is: "What is the most efficient way to compute a polynomial?". For example, the polynomial $xyz+2xy+xz+2x+yz+2y+z+2$ can be represented more compactly as $(1+x)\cdot(1+y)\cdot(2+z)$. Under the former representation the polynomial takes 8 multiplications and 7 additions to compute, while the latter offers a more efficient way of computation, taking only 3 additions and 3 multiplications.

Of course, the notion of efficiency as well as the computation model should be clarified before embarking on a serious investigation of the question, and we elaborate more on this later. However, what is apparent is that for any polynomial of interest, it is useful to *know* a threshold efficiency which cannot be surpassed by any algorithm computing the polynomial. Proving such a threshold, or *lower bound*, immensely helps one understand the polynomial as well as the algorithm (computing the polynomial). One can conclude that the best known algorithm *is* the best algorithm, if its efficiency and the proven lower bound happen to match. Even otherwise, i.e. if there is a gap between the two, algorithmists can inspect the proof and make informed decision on how to go about. Thus, as most problems in computer science do, polynomial computation offers two avenues of exploration for researchers, namely, proving a sufficiently high lower bound for a polynomial of interest, or coming up with a more-efficient-

than-current algorithm for the polynomial. Oftentimes the two processes are interdependent.

Several other questions, practically important ones, can be asked of polynomials, while keeping efficiency in mind. Consider for instance checking whether two polynomials (represented differently) are equal or not. This is equivalent to asking whether their difference, which is also a polynomial, is identically zero or not. Checking (efficiently) whether a polynomial (computed by a model) is identically zero is known as Polynomial Identity Testing (PIT). Its practical significance comes from the fact that many problems in computer science can be reduced to the problem of PIT. Another interesting problem is the *reconstruction problem*: from the knowledge of what a "black-box polynomial" evaluates to at certain inputs, find a representation (under the computation model agreed upon), preferrably an efficient one, of the polynomial. One can see that polynomial interpolation is a special case of the reconstruction problem. We remark that all the questions above are interdependent. To shed more detail on that, we need to elaborate on our computation model and the notion of efficiency.

An algebraic expression such as the ones we gave above (which contains just $+, -$ and $\cdot$ operators and paranthesis, no exponents) is called a *formula*. Formulas can be considered as a computational model. To attach a notion of efficiency to it, we first define the size of a formula as the number of $+, -$ and $\cdot$ operators, called the basic operations, in it. For a formula to qualify as *efficient*, we demand that the formula size be bounded by a function that is a polynomial in the number of variables. Notice however that in practice we tend to reuse intermediate results when possible. For example, while the formula $(x + y + z + 2) \cdot (x + y + z + 3)$ has size 7, a more efficient way seems to first compute $x + y + z$ and assign it to an fresh variable w and then compute $(w + 2) \cdot (w + 3)$. Even if we count the intermediate assignment as a basic operation we have in total $2 + 1 + 3 = 6 < 7$ operations. Formally, such a model is called an *algebraic straight line program* (ASLP). An ASLP is a sequence of instructions of the form $a = b \circ c$ where $\circ$ is one of $+, -$ and $\cdot$, a is a fresh variable and b and c are input variables or fresh-at-a-previous-instruction variables. The size of an ASLP is the number of instructions in it, and the efficiency is determined by the size of the ASLP in the same way as that for a formula. In Section 2.3 of Chapter 2 we define the model of *arithmetic circuits*. It is easy to see the equivalence between arithmetic circuits and algebraic straight line programs. Roughly, an arithmetic circuit is a directed acyclic graph with leaves corresponding to inputs and internal nodes corresponding to basic operations. Operands for a basic operation at an internal node come through the incoming edges and the result is made available on the outgoing edges. The size of an arithmetic circuit is the number of edges in the circuit.

Proving arithmetic circuit lower bounds and PIT are connected. To see how, we first remark that a randomized poly-time algorithm exists for PIT. Since it is widely believed that problems solvable in randomized poly-time are also solvable in deterministic poly-time (more precisely, $\mathsf{BPP} = \mathsf{P}$), a natural attempt is to derandomize PIT. Kabanets and Impagliazzo [KI04] showed that a subexponential time deterministic algorithm for PIT of arithmetic circuits implies either a superpolynomial lower bound for arithmetic circuits or $\mathsf{NEXP} \not\subset \mathsf{P/poly}$. In the reverse direction, they showed that a superpolynomial (similarly, exponential) lower bound for arithmetic circuits implies subexponential (similarly, quasipolynomial) time PIT. Agrawal [Agr05] showed that a polynomial time blackbox PIT algorithm implies a superpolynomial lower bound for circuits computing an explicit (PSPACE-computable) polynomial.

The PIT problem and the circuit reconstruction problem are connected as well. Using the test points given by the PIT algorithm we can distinguish two circuits, by evaluating their difference on those points. The techniques used for distinguishing two circuits have at times been used for designing some kind of learning algorithms for circuit reconstruction ([SY10]). In this way, proving arithmetic circuit lower bounds has implications on several areas of algebraic complexity theory.

Speaking of lower bounds, we would like to draw parallels with the boolean complexity theory. In fact, one of the motivations behind developing the theory of arithmetic circuits was to gain new insights on resolving the P vs NP problem. It is known that $\mathsf{VP} \neq \mathsf{VNP}$ implies $\mathsf{P} \neq \mathsf{NP}$ in the nonuniform setting, under the generalized Riemann hypothesis [Bür00]. Therefore it is plausible that proving $\mathsf{VP} \neq \mathsf{VNP}$ is easier. However, despite decades of effort this remains unresolved.

Some known lower bounds

In the course of time, several restricted models of arithmetic formulas have been considered and lower bounds are proven. Nisan [Nis91] considered the setting of noncommutative rings and proved an exponential lower bound on the size of (noncommutative) circuits computing (a noncommutative version of) determinant (Det_n). Similarly monotone circuits are considered. In monotone circuits negative constants and subtraction are prohibited throughout the computation. Jerrum and Snir proved an exponential $2^{\Omega(n)}$ lower bound on monotone circuits computing the permanent (Perm_n) of an $n \times n$ matrix.

Just like in the boolean world, arithmetic circuits with some structural restrictions have been studied. Constant depth and multilinearity are among those, and are relevant to this discussion. Under multilinearity constraint (first defined in [NW97]), the circuit is syntactically

forced to compute multilinear polynomials at all gates. (A polynomial is multilinear if its degree with respect to every variable is at most one.) Computation aspects of multilinear polynomials are of great interest for several reasons. To name a few, firstly, for every boolean function there is a straightforward multilinear polynomial that matches the boolean function on 0-1 values. Secondly, important polynomials like Det_n, Perm_n (which is VNP-complete), and the iterated matrix multiplication (IMM) are multilinear. Furthermore, the smallest known circuits computing Perm_n and IMM happen to be multilinear. The first nontrivial bound on multilinear models was by Raz [Raz09], who proved that any multilinear formula computing Det_n takes $n^{\Omega(\log n)}$ size. Subsequently [RY08, RY09] showed superpolynomial lower bounds (and separations) on constant depth (syntactically) multilinear circuits, against Det_n and a newly-defined multilinear polynomial. A natural question now is whether, against such multilinear polynomials, nontrivial bounds can be proved on models that subsume multilinear formulas and constant depth circuits. We shall return to this question later.

Our understanding of constant depth circuits (particularly depth three and depth four circuits) is relatively better. It is also where the distinction between the behavior arithmetic circuits and boolean circuits becomes stark. It started with [VSBR83], where they proved that any arithmetic circuit of size s computing a polynomial of degree d can be converted into one with depth $\log s \cdot \log d$, just at polynomial-blowup in size. This was taken forward by a series of works [AV08, Koi12, Tav13] to show that any arithmetic circuit of size s computing an N-variate polynomial f_N of degree d can be transformed into a depth four circuit of size $2^{\sqrt{d \cdot \log(d \cdot s) \cdot \log N}}$. In particular if s is subexponential in N (and $d = N^{O(1)}$) then so is the size of the depth four circuit. Therefore, to show a superpolynomial lower bound on general arithmetic circuits, it is sufficient to show a sufficiently high superpolynomial lower bound, i.e. $N^{\omega(\sqrt{d})}$, on depth four circuits computing some explicit polynomial. (Furthermore if the polynomial is in VNP then VP $\neq$ VNP follows.) In contrast, no such depth reduction is known for boolean circuits.

Gupta et al. [GKKS13] took a step further in the course of depth reduction. [GKKS13] and [Tav13] together imply that any arithmetic circuit of size s computing a polynomial f_N of degree d can be transformed into a depth three circuit of size $2^{\sqrt{d \cdot \log(d \cdot s) \cdot \log N}}$, although they require the underlying field to be of characteristic zero. Also, in the circuit resulted from a reduction to depth three, the polynomials computed at intermediate gates can be of very high degree compared to the degree of the output polynomial. In other words, reduction to depth three does not preserve *homogeneity*. A circuit is said to be homogeneous if all its gates compute homogeneous polynomials, i.e. polynomials in which all the monomials are of the same degree. The reduction to $\log s \cdot \log d$-depth and to depth four, mentioned before, preserves homogeneity.

In this light, proving an $N^{\omega(\sqrt{d})}$ for depth three or depth four circuits is a plausible goal, while homogeneity can be assumed in the case of depth four. On depth four homogeneous circuits, a series of works by [KLSS] and [KS14] yielded an $N^{\Omega(\sqrt{d})}$ lower bound for IMM.

However, for depth three circuits, the issue of high formal degree seems to be a hurdle for existing proof techniques. Most of the current proofs work when the formal degree of the circuit (i.e. the maximum degree of intermediate polynomials potentially computed) does *not* exceed the number of underlying variables N. (A few notable exceptions are [KS15a, KS15b, KS16, KST16a].) Note that this is in agreement with the aforementioned fact that we know some nontrivial lower bounds on multilinear models: multilinearity, by definition, restricts the formal degree to be at most N. A step forward towards understanding the high-formal-degree regime could be to generalize the multilinear model. (Another direction would be to homogenize a formula without a signficant blow up in size – this would bring down the formal degree. However no efficient formula homogenization is known. In contrast, efficient circuit homogenization is known.)

With this motivation, Kayal and Saha [KS15c] defined *multi-r-ic* formulas. In a multi-r-ic formula the intermediate polynomials can have formal degree as high as r times the number of variables. Clearly, multilinear formulas are the $r = 1$ case of multi-r-ic formulas.

1.1 Previous Works

[KS15c] proved a $2^{\Omega(N/2^{25 \cdot r})}$ lower bound on depth three multi-r-ic circuits computing a multi-r-ic (not multilinear) polynomial. This was improved to $2^{\Omega(N)}$ by Kayal, Saha and Tavenas [KST16b]. [KST16b] also showed that a polynomial computed by a multi-r-ic formula of depth three is "hard" for multi-r-ic homogeneous formulas of arbitrary depth. Finally, they proved lower bounds on multi-r-ic formulas of depth three and depth four, against certain multilinear and non-multilinear polynomials. The underlying hope is, techniques used to prove depth three and depth four multi-r-ic formula lower bounds will shed some light on general multi-r-ic fomulas just like in the multilinear ($r = 1$) case – consider for instance the proof of multilinear formula lower bound using log-product formula [RY09], the latter is a kind of multilinear depth four formula. Here we only list the [KST16b] results against multilinear polynomials. With Det_n as the target polynomial, they showed a $2^{\Omega(n/r)}$ lower bound on multi-r-ic depth three circuits and a $2^{\Omega(\sqrt{n}/r)}$ lower bound on multi-r-ic depth four circuits. With $\mathrm{IMM}_{n,d}$ as the target polynomial (which is the $(1,1)$-th entry of the product of d symbolic matrices of size $n \times n$ each), they showed a lower bound of $\left(\frac{n}{r}\right)^{\Omega(d)}$ and $\left(\frac{n}{r}\right)^{\Omega(\sqrt{d/r})}$ on multi-r-ic formulas of depth three and depth four respectively.

We note that the bounds against $\mathrm{IMM}_{n,d}$ are functions of degree d (and the number of variables $N \approx n^2 \cdot d$). Since a multilinear polynomial has at most $\binom{N}{d} \approx \left(\frac{N}{d}\right)^d$ monomials, for $r = 1$ (or $r \ll d$) we can say that the bound is close to the optimum in the depth three case and close to $\left(\frac{N}{d}\right)^{\sqrt{d}}$ in the depth four case (for $d \leq N^{0.9}$, say).

It is also relevant to talk about the proof techniques used in the recent lower bound proofs. Many circuit lower bound results in the literature have followed a common template. First, the circuit under study is brought into the form of (unless it already is) a sum of "building blocks", i.e. smaller circuits with possibly a certain structural property. Their number is ensured to be not much more than the original circuit size. Next, a subadditive function that maps polynomials to numbers is defined. Such a function is referred to as a *measure* in the literature. On one hand, we want to devise a measure such that (the polynomial computed by) a building block has low measure. On the other hand, we want to come up with an explicit polynomial that has high measure. Finally, if such a polynomial is computed by our circuit then the circuit must have a large measure which in turn demands a very large number of building blocks and thus a large size.

Works such as [Nis91, NW97] showed that the dimension of the space of partial derivatives often serves as a good measure. Building upon this concept, Kayal [Kay12] introduced The *shifted partials dimension* (SPD) measure. Roughly speaking, SPD of a polynomial f is the dimension of the space of all k-order partial derivatives (of f) multiplied by all possible monomials of degree at most ℓ, where the parameters k, ℓ are integers. (In Chapter 2 we define SPD and roughly deduce the optimal choices of k and ℓ.) [GKKS14] used SPD to prove a $2^{\Omega(\sqrt{n})}$ lower bound on depth four homogeneous circuits with bottom fanin bounded by $\sqrt{n}$, computing Det_n. An $N^{\Omega(d)}$ lower bound for homogeneous depth four circuits (without the bottom fan-in restriction) was proved in [KLSS] by introducing a variant of the SPD measure, called the (dimension of) *projected shifted partials*. Subsequent works such as [KS14, KS15a, KS15b, KS16] used the projected shifted partials measure to exploit the structure of the models they were trying to size-lower-bound.

[KST16b], keeping multi-r-ic models in mind, introduced another variant of SPD and called it the (dimension of) *shifted skewed partials* (SSP). SSP differs from SPD in that it considers k-th order derivatives with respect to a subset $\mathbf{y}$ (say) of variables and considers the derivatives to be multiplied by *non*-$\mathbf{y}$ *monomials* of degree at most ℓ. Furthermore, the remnant $\mathbf{y}$-variables (after taking derivatives) are "killed" by setting to zero. The term *skew* stands for the (asymptotic) size disparity between $\mathbf{y}$ and its complement (where the larger subset is $\mathbf{y}$), which [KST16b] found to give an edge (over SPD) when the target polynomial is IMM (a VP

polynomial) of low degree.

1.2 Our results

While [KST16b] show a nontrivial lower bound on depth four multi-r-ic circuits that holds for a range of d's (in particular, low d's), we give a lower bound on the same model that remains superpolynomial for a wider range of r. (For more comparisons, see Section 1.3.)

Theorem 1 (Multi-r-ic depth four)**.** *Let N, d, r be positive integers such that $0.51 \cdot N \leq d \leq 0.9 \cdot N$ and $r \leq (N \cdot \log N)^{0.9}$. Then there is an explicit N-variate degree-d multilinear polynomial in* VNP *such that any multi-r-ic depth four circuit computing it has size $2^{\Omega\left(\sqrt{\frac{N \cdot \log N}{r}}\right)}$.*

Like in previous works, the route to prove the above theorem is via reduction to a restricted model of depth four circuits where the bottom-support (i.e. the maximum number of variables feeding to a bottom layer multiplication gate) is bounded.

Theorem 2 (Multi-r-ic τ-bottom-support depth four)**.** *Let N, d, r, τ be positive integers such that $2^{21} \cdot \log N \leq d \leq 0.9 \cdot N$, and $\frac{2^{21}}{5000} \cdot \log N \leq \tau \cdot r \leq \frac{d}{5000}$. Then there is an explicit N-variate degree-d multilinear polynomial in* VNP *such that any τ-bottom-support multi-r-ic depth four circuit computing it has size at least $\left(\frac{\tau^{20} \cdot d}{N \cdot r}\right)^{\left(\frac{0.00001 \cdot d}{\tau \cdot r}\right)}$.*

We use the same strategy as in [KST16b] to prove Theorem 2. However, for high degree range (which is chosen to maximize the lower bound as a function of N) SSP does not seem to do any better than SPD, and hence we choose SPD as the measure. What gives us leverage is our choice of the *Nisan-Wigderson polynomial* (defined in [KSS14, KST16a]) as the target polynomial, which is in VNP (in comparison to [KST16b]'s choice of IMM, a VP polynomial).

To build some intuition we separately prove a lower bound on multi-r-ic depth three circuits (see Section 1.3 for another motivation).

Theorem 3 (Multi-r-ic depth three)**.** *Let N, d, r be positive integers such that $2^{24} \cdot r^{1.1} \cdot \log N \leq d \leq 0.9 \cdot N$. Then there is an explicit N-variate degree-d multilinear polynomial in* VNP *such that any multi-r-ic depth three circuit computing it has size $2^{\Omega\left(\frac{d}{r^{1.1} \cdot \log N}\right)}$.*

Remarks.

1. The constant 0.9 in the above theorem statements can be brought arbitrarily closer to 1, and the results still hold with slightly changed constants in the exponents.

2. By slightly tweaking the proof, the bound in Theorem 3 can be changed to $2^{\Omega\left(\frac{d^{1.1}}{r^{1.1} \cdot N^{0.1}}\right)}$, which is better than $2^{\Omega\left(\frac{d}{r^{1.1} \cdot \log N}\right)}$ for $d = \Theta(N)$.

1.3 Discussions

In this section we compare our results with [KST16b]'s. For summary, see Table 1.1. (N and d denote the number of underlying variables and the degree of the polynomial computed, respectively.)

Table 1.1: Comparison of our results with [KST16b]

Multi-r-ic Model	**Work**	**Lower Bound** (LB)	**Constraints**	**Range of r for $N^{\omega(1)}$ LB**
Depth four	[KST16b]	$\left(\frac{N}{d\cdot r^2}\right)^{\Omega(\sqrt{\frac{d}{r}})}$	$\log^2 \frac{N}{d} \leq d$	$r \leq N^{1/3}$ for $d = \Theta(N^{1/3})$
	Ours	$2^{\Omega(\sqrt{\frac{N\cdot\log N}{r}})}$	$0.51 \cdot N \leq d \leq 0.9 \cdot N$ $r \leq (N \cdot \log N)^{0.9}$	$r \leq (N \cdot \log N)^{0.9}$
Depth three	[KST16b]	$\left(\frac{N}{d\cdot r^2}\right)^{\Omega(d)}$		$r \leq N^{1/2}$
	Ours	$2^{\Omega(\frac{d}{r^{1.1}\cdot\log N})}$	$2^{24} \cdot r^{1.1} \cdot \log N \leq d \leq 0.9 \cdot N$	$r \leq N^{0.9}$ for $d = \Theta(N)$
Depth four (τ-bottom-support)	[KST16b]	$\left(\frac{N}{d\cdot r^2}\right)^{\Omega(\frac{d}{\tau\cdot r})}$	$\log N \leq \tau \cdot r = o(d)$	$r \leq \left(\frac{N}{\tau}\right)^{1/3}$ for $d = \Theta((N \cdot \tau^2)^{1/3})$
	Ours	$\left(\frac{\tau^{20}\cdot d}{r\cdot N}\right)^{\Omega(\frac{d}{\tau\cdot r})}$	$\log N \leq \tau \cdot r = o(d)$	$r \leq \frac{N}{\tau}$ (for $d = \Theta(N)$ and $\tau \geq N^{1/21}$)

For multi-r-ic depth four lower bound.

1. Better range on r. In [KST16b], a lower bound of $\left(\frac{N}{d\cdot r^2}\right)^{\Omega\left(\sqrt{\frac{d}{r}}\right)}$ was shown for multi-r-ic depth four circuits computing $\mathrm{IMM}_{n,d}$ where $N \approx n^2 \cdot d$. For the bound to remain superpolynomial, r can be at the most $\min(\sqrt{\frac{N}{d}}, d)$. The expression $\min(\sqrt{\frac{N}{d}}, d)$ is maximized at $d = N^{1/3}$,

and hence r has to be less than $N^{1/3}$. On the other hand we show a lower bound of $2^{\Omega\left(\sqrt{\frac{N\cdot\log N}{r}}\right)}$ for $d \in [0.51 \cdot N, 0.9 \cdot N]$ and $r \leq (N \cdot \log N)^{0.9}$ which continues to remain superpolynomial in this range for r.

2. Improved lower bound. For any fixed function $r = r(N)$, [KST16b]'s lower bound of $\left(\frac{N}{d\cdot r^2}\right)^{\Omega\left(\sqrt{\frac{d}{r}}\right)}$ is maximized (as a function of N) to $2^{\Omega\left(\sqrt{\frac{N}{r^3}}\right)}$ at $d = \Theta\left(\frac{N}{r^2}\right)$. In comparison, Theorem 1 shows a lower bound of $2^{\Omega(\sqrt{\frac{N\cdot\log N}{r}})}$ which is asymptotically better as a function of N.

3. Extending the result of Raz and Yehudayoff ([RY09]). The best known lower bound for multilinear ($r = 1$) depth four circuits is $2^{\Omega(\sqrt{N\cdot\log N})}$ [RY09]. Our result can be seen as an extension of this lower bound to multi-r-ic depth four circuits, although the proof techniques in [RY09] and in here are quite different. In particular, [RY09] used rank of a partial derivatives matrix as the measure whereas we use SPD.

For multi-r-ic low-bottom-support depth four lower bound.

1. Better range on r. In [KST16b] a lower bound of roughly $\left(\frac{N}{d\cdot r^2}\right)^{\Omega\left(\frac{d}{\tau\cdot r}\right)}$ was shown for multi-r-ic depth four circuits with bottom support bounded by τ computing an N-variate degree-d polynomial family in VP. For the bound to remain superpolynomial, r can be at most $\min(\sqrt{\frac{N}{d}}, \frac{d}{\tau})$. Choosing $d = (N \cdot \tau^2)^{1/3}$ for any fixed function $\tau = \tau(N)$ to maximize $\min(\sqrt{\frac{N}{d}}, \frac{d}{\tau})$, we can choose r at most $\left(\frac{N}{\tau}\right)^{1/3}$. On the other hand, in Theorem 2, r can be chosen to be $\frac{N}{\tau}$ for $d = \Theta(N)$ and $\tau \geq N^{1/21}$. This gives a better range of r for higher τ.

2. Improved lower bound. The lower bound $\left(\frac{N}{d\cdot r^2}\right)^{\Omega\left(\frac{d}{\tau\cdot r}\right)}$ in [KST16b], for any fixed functions $r = r(N)$ and $\tau = \tau(N)$, is maximized (as a function of N) to $2^{\Omega\left(\frac{N}{\tau\cdot r^3}\right)}$ at $d = \Theta(\frac{N}{r^2})$. In comparison, Theorem 2 shows a lower bound of $\left(\frac{N^{0.9}}{r}\right)^{\Omega\left(\frac{N}{r\cdot\tau}\right)}$ for $d = \Theta(N)$ and $\tau \geq N^{1/21}$ which is asymptotically better as a function of N and τ, for higher τ.

For multi-r-ic depth three lower bound.

1. Better range on r. In [KST16b] a lower bound of $\left(\frac{N}{d\cdot r^2}\right)^{\Omega(d)}$ was shown for multi-r-ic depth three circuits computing $\mathrm{IMM}_{n,d}$ where $N \approx n^2 \cdot d$. This means that r can be at most $\sqrt{N/d} \leq \sqrt{N}$. On the other hand we show a lower bound of $2^{\Omega(\frac{d}{r^{1.1}\cdot\log N})}$ which remains superpolynomial for $r \leq N^{0.9}$ at $d = \Theta(N)$.

2. Shifted partials of elementary symmetric polynomials. [FLMS15] proved a lower bound on SPD of the elementary symmetric polynomial ESym_N^d where N and d are the number of variables and the degree respectively.

Theorem 4 ([FLMS15]). *Let $N, d, k \in \mathbb{N}$ be such that $d \leq \frac{\log N}{10 \cdot \log\log N}$ and $k = \lfloor \frac{d}{t+1} \rfloor$ for some $t \in \mathbb{N}$ satisfying $t = 1 \mod 4$. Then, for $\ell = \lfloor N^{1-1/(2\cdot t)} \rfloor$*

$$\mathsf{SPD}_{k,\ell}(\mathrm{ESym}^d_N) \geq \frac{(1-o(1)) \cdot \binom{N+\ell}{N} \cdot \binom{N-\ell}{k}}{\left(\frac{3\cdot\sqrt{N}}{2}\right)^k \cdot (2\cdot d)^t}.$$

Now $\binom{N}{k} \cdot \binom{N+\ell}{N} = \mathsf{SPD}^{\max}_{k,\ell}$ (say) is the maximum value of the shifted partials dimension of any N-variate multilinear polynomial we can get (see Section 2.5). Let us express the [FLMS15] lower bound as a fraction of $\mathsf{SPD}^{\max}_{k,\ell}$.

$$\begin{aligned}
\mathsf{SPD}_{k,\ell}(\mathrm{ESym}^d_N) &\geq \frac{0.9 \cdot \binom{N+\ell}{N} \cdot \binom{N}{k} \cdot \binom{N-\ell}{k}}{\left(\frac{3}{2}\cdot\sqrt{N}\right)^k \cdot (2\cdot d)^t \cdot \binom{N}{k}} \\
&\geq \frac{0.9 \cdot \binom{N-\ell}{k}}{(\frac{9}{4}\cdot N)^{k/2} \cdot (2\cdot d)^t \cdot \binom{N}{k}} \cdot \mathsf{SPD}^{\max}_{k,\ell}. \qquad (1.1)
\end{aligned}$$

For $t = \sqrt{d}$ (which is a typical value for t in the literature) we get $k \approx \sqrt{d} = t$ and $(2\cdot d)^t \approx (2\cdot d)^k \leq (\log N)^k$. The ratio $\frac{\binom{N-\ell}{k}}{\binom{N}{k}}$ can be bounded as below.

$$\begin{aligned}
\frac{\binom{N-\ell}{k}}{\binom{N}{k}} &= \frac{(N-\ell)\cdot(N-\ell-1)\ldots(N-\ell-k+1)}{N\cdot(N-1)\ldots(N-k+1)} \\
&= \frac{N-\ell}{N} \cdot \frac{N-\ell-1}{N-1} \cdots \frac{N-\ell-k+1}{N-k+1} \\
&= \left(1-\frac{\ell}{N}\right)\cdot\left(1-\frac{\ell}{N-1}\right)\ldots\left(1-\frac{\ell}{N-k+1}\right) \\
&\geq \left(1-\frac{\ell}{N-k+1}\right)^k \\
&\geq e^{-\frac{2\cdot k\cdot \ell}{N-k+1}} && \text{(as } \ell = o(N) \text{ for } d \leq \tfrac{\log N}{10\cdot\log\log N} \text{ in Theorem 4;} \\
& && \text{use Bound 2 from Section 2.2)} \\
&\geq e^{-2\cdot k} && \text{(again, as } \ell = o(N)\text{).}
\end{aligned}$$

Plugging this in Equation (1.1) we get

$$\mathsf{SPD}_{k,\ell}(\mathrm{ESym}^d_N) \geq \frac{0.9\cdot\mathsf{SPD}^{\max}_{k,\ell}}{(\frac{9}{4}\cdot N)^{k/2}\cdot(\log N)^k \cdot e^{2\cdot k}} \geq \frac{0.9\cdot\mathsf{SPD}^{\max}_{k,\ell}}{\left(\sqrt{\frac{9}{4}\cdot N}\cdot\log N \ \cdot e^2\right)^k} \geq \frac{1}{N^{0.51\cdot k}}\cdot\mathsf{SPD}^{\max}_{k,\ell}.$$

Thus $\mathsf{SPD}_{k,\ell}(\mathrm{ESym}_N^d)$ is at most a factor of $\frac{1}{N^{0.51\cdot k}}$ away from $\mathsf{SPD}_{k,\ell}^{\max}$ for the choice of low d and ℓ as above.. Is $\mathsf{SPD}_{k,\ell}(\mathrm{ESym}_N^d)$, however, actually a factor $\frac{1}{N^{\gamma\cdot k}}$ away (for some constant γ) from $\mathsf{SPD}_{k,\ell}^{\max}$, or is it "much closer" to $\mathsf{SPD}_{k,\ell}^{\max}$? We observe that in a certain sense it is the former case (at least when d is high and ℓ is chosen appropriately) as explained below:

In Section 3.5 we show that the shifted partials dimension of a multilinear depth three circuit with top fan-in[1] s, for the "optimum" choice of ℓ,[2] is at most $s \cdot \left(\frac{33\cdot N^{1/11}\cdot k^{1/1.1}}{\alpha\cdot d}\right)^k \cdot \mathsf{SPD}_{k,\ell}^{\max}$ where $0 < \alpha < 0.5$ is a constant. There is a multilinear depth three circuit with top fanin $s = N+1$ that can compute ESym_N^d, due to Ben-Or [SW01]. Therefore

$$\mathsf{SPD}_{k,\ell}(\mathrm{ESym}_N^d) \leq \ (N+1)\cdot\left(\frac{33\cdot N^{1/11}\cdot k^{1/1.1}}{\alpha\cdot d}\right)^k \cdot \mathsf{SPD}_{k,\ell}^{\max}.$$

Setting $k = \sqrt{d}$ and $d \geq N^{0.17}$ we get

$$\mathsf{SPD}_{k,\ell}(\mathrm{ESym}_N^d) = O\left(\frac{1}{N^{0.001\cdot k}}\right)\cdot \mathsf{SPD}_{k,\ell}^{\max}.$$

Thus, in this particular sense, our upper bound "complements" the lower bound shown in [FLMS15]. Indeed, it is this gap between $\mathsf{SPD}_{k,\ell}(\mathrm{ESym}_N^d)$ and $\mathsf{SPD}_{k,\ell}^{\max}$ that helps us prove the lower bound in Theorem 3. We note however that the [FLMS15] result holds for small values of degree, i.e. $d \leq \frac{\log N}{10\cdot\log\log N}$ whereas our result holds for the degree range $N^{0.17} \leq d \leq 0.9\cdot N$. Also, we have chosen a different value for ℓ.

We would also like to note that this loss of $1/N^{0.001\cdot k}$ factor does *not* imply so far that it is not possible to prove a $N^{\Omega(\sqrt{d})}$ lower bound for homogeneous depth four circuits with low bottom fanin computing ESym_N^d for higher values of d (using SPD as the measure).[3] This is because, in principle (as explained in Section 2) the best possible lower bound achievable using $\mathsf{SPD}_{k,\ell}^{\max}$ is $\binom{N}{k}^{1-\epsilon} \geq \left(\frac{N}{k}\right)^{(1-\epsilon)\cdot k}$ which for the setting of parameters $k = \sqrt{d}$, $d = N^{0.52}$ and $\epsilon = 0.1$ is at least $N^{0.6\cdot k}$. A loss of $\frac{1}{N^{0.001\cdot k}}$ factor would still give a $N^{\Omega(\sqrt{d})}$ lower bound.

A common motivation.

Another motivation common to both the depth three and depth four lower bound results is as follows. As mentioned before, [KST16b] used shifted skewed partials, a variant of SPD, as the measure. We wondered if it is possible to show lower bounds for multi-r-ic depth three and

[1] The top fan-in of a circuit is the in-degree of its root.

[2] In Section 2.5 we explain what the optimum choice of ℓ is.

[3] [FLMS15] already proves such a lower bound but for low d, i.e. $d \leq \frac{\log N}{10\cdot\log\log N}$.

depth four circuits using SPD (as is in [Kay12]) primarily because this measure has the nice property of *invariance under affine transformations* which is lacking for shifted skewed partials. Indeed our proofs show that this is possible when degree is high (although for low degree d and expressing the lower bound as a function of N and d which is the case covered in [KST16b], the variant measure, shifted skewed partials, seems to be important). This means our lower bound in Theorem 3 continues to hold for any depth three circuit that is derived from a multilinear depth three circuit by replacing each variable by an affine form. A similar generalization is also true in Theorem 2. However, to prove Theorem 1, we use random restriction on the circuit to reduce it to a circuit with low bottom support. This step does not carry through if the variables of the circuit are replaced by affine forms. So, applying affine invariance property to Theorem 1 does not seem to give a lower bound for a more general model than what is considered in the theorem.

1.4 Outline of the rest of the thesis

The rest of the thesis is dedicated to proving the three theorems stated above. In Chapter 2, Preliminaries, we set up notations, define the SPD measure and discuss its properties. Chapter 3 proves Theorem 3 while postponing the discussion of the target polynomial. Similarly, Chapter 4 proves Theorem 2. The target polynomial is elaborated in Chapter 5. The main theorem, Theorem 1, is finally proven in Chapter 6. We conclude with remarks on future work, in Chapter 7.

Chapter 2

Preliminaries

2.1 Notations

We use a bold letter, like $\mathbf{x}, \mathbf{y}$ etc., to denote a set of variables. Elements of $\mathbf{x}$ are denoted by $x_1, x_2, \ldots$ etc. and are called $\mathbf{x}$-variables. An $\mathbf{x}$-monomial is a monomial only containing $\mathbf{x}$-variables. On the other hand, by an f-monomial, where f is a polynomial, we mean a monomial whose coefficient in f is nonzero. If μ is an $\mathbf{x}$-monomial and ν is a $\mathbf{y}$-monomial then by the $\mathbf{x}$-part (respectively $\mathbf{y}$-part) of $\mu \cdot \nu$ we mean μ (respectively ν). For a polynomial f, $\deg_x f$ denotes the degree of f with respect to a variable x, and $\deg f$ denotes the total degree of f. For an integer $\ell \geq 0$, the set of all $\mathbf{x}$-monomials of total degree at most ℓ is denoted by $\mathbf{x}^{\leq \ell}$. For two sets F and G of polynomials, $F \cdot G$ (respectively, $F + G$) will denote the set of products (respectively, sums) of two polynomials one from F and G each.

Let X be a set and $a \leq b$ be two integers. The set of all subsets of X of size between a and b (inclusive) is denoted by $\binom{X}{[a,b]}$, and simply by $\binom{X}{a}$ when $a = b$. The set of positive integers is denoted by $\mathbb{N}$ and the set $\{1, 2, \ldots, n\}$ is denoted by $[n]$. 'log' and 'ln' denote logarithms to base 2 and base e respectively. Sometimes we use the term $\mathsf{poly}(n)$ to mean $n^{O(1)}$. In statements and proofs of Theorems 1,2 and 3, we assume N, the number of variables, to be sufficiently large (so as to legitimize inequalities that hold asymptotically). Also, sometimes we omit floor ($\lfloor\ \rfloor$) and ceil ($\lceil\ \rceil$) notations for integer-valued functions of N, d etc. for simplicity of presentation.

2.2 Some well-known bounds

The following estimates will be useful in the upcoming chapters.

1. **Exponential upper bound.** For a real number x, $1 + x \leq e^x$.
2. **Exponential lower bound.** For a real number $0 < x < \frac{1}{2}$, $1 - x \geq e^{-2 \cdot x}$.

3. **Binomial bounds.** For integers $0 \leq k \leq n$, $\left(\frac{n}{k}\right)^k \leq \binom{n}{k} \leq \left(\frac{e \cdot n}{k}\right)^k$.

We also use Chernoff bound.

4. **Chernoff bound.** Let X be the sum of several independent 0-1 random variables. Let $\mu = E[X]$. Then for any constant $\varepsilon > 0$,

$$\Pr[|X - \mu| \geq \varepsilon \cdot \mu] \leq 2 \cdot e^{-\varepsilon^2 \cdot \mu/3}.$$

2.3 Arithmetic circuits

An *arithmetic circuit* is a directed acyclic graph in which every node with in-degree 0 (called *input gate*) is labelled with a variable or a field element, and every node with positive in-degree is labelled with either '+' (in which case the node is a *sum gate*) or '×' (in which case the node is a *product gate*). If there is an edge from a node u to a node v then u is called a *child* of v. With every node we associate a polynomial and say that the node *computes* the polynomial, as follows: An input gate is said to compute what it is labelled with. A sum (respectively product) gate is said to compute the sum (respectively product) of the polynomials associated with its children. We consider circuits which have exactly one *root*, i.e. the node with out-degree 0, and a circuit is said to compute the polynomial its root computes. Also, we allow edges to be labelled with field constants. If an edge from node u to node v is labelled with a constant α and u is computing a polynomial f then v considers $\alpha \cdot f$, rather than mere f, as the input coming from u.

The *size* of a circuit is the number of edges in it. The *depth* of a circuit is the length of the longest path from an input gate to the root. While size captures the number of times the basic operations are executed, depth captures the complexity of parallel computation: given sufficiently many processors, it is the number of parallel steps taken to compute the output.

An arithmetic circuit in which all nodes have out-degree at most one is called a *formula*. When the depth is constant, we use terms circuit and formula interchangeably, as in that case a circuit can be converted into a formula with just poly-size blow up.

Depth three and depth four circuits. An expression that is a sum of products of linear polynomials clearly corresponds to an arithmetic circuit of depth three. From now on, by a depth three circuit we mean such an expression. Such circuits are also called $\Sigma\Pi\Sigma$ circuits, meaning the circuit has a top sum gate followed by a layer of product gates and finally a bottom layer of sum gates. Similarly a circuit with a sum gate on top, followed by a layer of product

gates, then a layer of sum gates again, and finally a bottom layer of product gates corresponds to a depth four circuit (also called a $\Sigma\Pi\Sigma\Pi$ circuit). Naturally, a depth four circuit is associated with an expression that is a sum of product of sum of monomials, and hence by a depth four circuit we mean such an expression. Further if the monomials computed at the bottom layer of product gates of a depth four circuit are such that each of them has at most τ variables appearing in it, then we say that the depth four circuit has τ-*bottom-support*.

Multi-r-ic formulas. The *formal degree* of an input gate g with respect to a variable x is defined to be 1 if g is labelled with x and 0 if g is labelled with a different variable or a field element. The formal degree of a sum (respectively product) gate g with respect to a variable x is defined to be the maximum (respectively sum) of the formal degrees of its children with respect to x. Let r be a positive integer. A *multi-r-ic* formula is an arithmetic formula such that every gate in it has formal degree at most r with respect to every variable. If $r = 1$, a multi-r-ic formula is called a *multilinear* formula. A polynomial is said to be multilinear if the degree of every variable is at most one in every monomial of the polynomial.

Homogeneous circuits. A polynomial is said to be homogeneous if all its monomials are of the same degree. A circuit is said to be homogeneous if all its gates compute homogeneous polynomials.

2.4 Arithmetic complexity classes

Valiant [Val79] defined arithmetic complexity classes VP and VNP, analogous to the noted boolean complexity classes P and NP. In the definitions below, $\mathbb{F}$ denotes a field.

Definition 1. *A family of polynomials $\{f_n\}$ over $\mathbb{F}$ is said to be* p-bounded *if there is a polynomial $t : \mathbb{N} \to \mathbb{N}$ such that for every n, f_n has at most $t(n)$ variables, has degree at most $t(n)$ and can be computed by a circuit of size at most $t(n)$. The class of all p-bounded families over $\mathbb{F}$ is denoted by $\mathsf{VP}_{\mathbb{F}}$, or simply, VP.*

Definition 2. *A family of polynomials $\{f_n\}$ over $\mathbb{F}$ is said to be* p-definable *if there is a polynomial family $\{g_n\}$ over $\mathbb{F}$ in VP and polynomials $t, k : \mathbb{N} \to \mathbb{N}$ such that*

$$f(x_1, \ldots, x_{t(n)}) = \sum_{(w_1,\ldots,w_{k(n)}) \in \{0,1\}^{k(n)}} g(x_1, \ldots, x_{t(n)}, w_1, \ldots, w_{k(n)}).$$

The class of all p-definable families over $\mathbb{F}$ is denoted by $\mathsf{VNP}_{\mathbb{F}}$, or simply, VNP.

In fact, in the above definition it can be assumed that g is computable by a $\mathsf{poly}(n)$ sized formula (instead of a circuit). Valiant [Val79] gave a sufficient condition for a polynomial

family's membership in VNP. Called *Valiant's criterion*, it turns out to be useful in Chapters 5 and 6. We state it below and prove a special case of it where multilinearity and binary coefficients are assumed. (Polynomial families showing up in the later chapters are multilinear and have binary coefficients.)

Proposition 5 (Valiant's criterion)**.** *A family $\{f_n\}$ of multilinear polynomials with 0-1 coefficients is in* VNP *if there is a* $\mathsf{poly}(n)$*-time algorithm*[1] *which, given any monomial, computes its coefficient in f_n.*

Proof. Let $x_1, \ldots, x_{t(n)}$ be the variables of f_n for some polynomial t. It is given to us that there is a deterministic turing machine M that takes as input a binary vector $\mathbf{e}$, corresponding to the monomial $x_1^{e_1} \ldots x_{t(n)}^{e_{t(n)}}$, runs in time $u(n)$ for some polynomial u, and outputs the monomial's coefficient in f_n. Along the lines of the proof of Cook-Levin theorem one can construct a 'small' boolean formula ϕ that simulates M on $\mathbf{e}$. ϕ obtained in this way is in variables $\mathbf{e} \cup \mathbf{y}$ such that $|\mathbf{e} \cup \mathbf{y}| = O(u^2(n))$ (where $\mathbf{y}$-variables are fresh), has size $\mathsf{poly}(n)$, and has the following property: For every assignment $\mathbf{a} \in \{0,1\}^{|\mathbf{e}|}$,

1. if $M(\mathbf{a}) = 0$ then $\phi(\mathbf{a}, \mathbf{y})$ is not satisfiable, and
2. if $M(\mathbf{a}) = 1$ then there is exactly one assignment $\mathbf{b} \in \{0,1\}^{|\mathbf{y}|}$ such that $\phi(\mathbf{a}, \mathbf{b}) = 1$.

We *arithmetize* ϕ. In other words, we construct an arithmetic formula g_n in variables $\mathbf{e} \cup \mathbf{y}$ such that it agrees with ϕ on inputs from $\{0,1\}^{|\mathbf{e}|+|\mathbf{y}|}$. This is easily done by replacing expressions of the form $\neg z_1$ with $(1 - z_1)$, $z_1 \wedge z_2$ with $z_1 \cdot z_2$, and $z_1 \vee z_2$ with $z_1 + z_2 - z_1 \cdot z_2$. It is easy to see that g_n as a polynomial family is p-bounded: Firstly g_n has just $O(u^2(n))$, i.e. $\mathsf{poly}(n)$, variables. Without loss of generality we could assume ϕ to be in 3CNF with $\mathsf{poly}(n)$ clauses. In that case the degree of a clause (after arithmetization) would be at most a constant and hence the degree of g_n would be $\mathsf{poly}(n)$. Finally, g_n as a formula is of size at most three times that of ϕ, which is $\mathsf{poly}(n)$ again. With the p-boundedness of g_n established, proving the equality below is sufficient to show f_n is VNP:

$$f_n(x_1, \ldots, x_{|\mathbf{e}|}) = \sum_{\substack{\mathbf{a} \in \{0,1\}^{|\mathbf{e}|} \\ \mathbf{b} \in \{0,1\}^{|\mathbf{y}|}}} g_n(\mathbf{a}, \mathbf{b}) \cdot \prod_{1 \le i \le |\mathbf{e}|} x_i^{a_i}.$$

Proof is by comparing coefficients on both sides. The coefficient of a fixed monomial $\prod_{1 \le i \le |\mathbf{e}|} x_i^{a_i}$ on the right hand side (RHS) is $\sum_{\mathbf{b} \in \{0,1\}^{|\mathbf{y}|}} g_n(\mathbf{a}, \mathbf{b})$. If the monomial's coefficient on the left hand

[1] In fact, even the much weaker condition of being able to compute the coefficient of a given monomial of f_n in #P suffices to show that the family $\{f_n\}$ is in VNP.

side (LHS) is 0 then ϕ is unsatisfiable, i.e. $g_n(\mathbf{a}, \mathbf{b}) = 0$ for all $\mathbf{b} \in \{0,1\}^{|\mathbf{y}|}$ and thus the coefficient on the RHS is 0. On the other hand if the coefficient on the LHS is 1 then there is exactly one satisfying $\mathbf{b} \in \{0,1\}^{|\mathbf{y}|}$ for ϕ, and hence the coefficient on the RHS is 1 as well. □

In the rest of the text whenever we make statements like "an N-variate polynomial f is computed by a circuit C," we mean that there is a *family* of polynomials $\{f_n\}$ and a *family* of circuits $\{C_n\}$ with $N = \mathsf{poly}(n)$, such that f_n is computed by C_n for $n \in \mathbb{N}$. Statements like "a polynomial is in VNP" should also be similarly interpreted.

2.5 The shifted partials dimension measure

Let $\mathbb{F}$ be a field. For integer parameters $k, \ell \geq 0$, the shifted partials dimension is a function $\mathsf{SPD}_{k,\ell} : \mathbb{F}[\mathbf{x}] \to \mathbb{N}$ defined as follows. Let $f \in \mathbb{F}[\mathbf{x}]$. For any $\mu \in \binom{\mathbf{x}}{k}$, we write $\partial_\mu f$ to denote $\frac{\partial^k f}{\partial \mu_1 \cdot \partial \mu_2 \cdot \ldots \cdot \partial \mu_k}$ where $\mu_1, \ldots, \mu_k$ are elements of μ. Also, for a set of polynomials S we denote by $\partial_\mu S$ the set $\{\partial_\mu f : f \in S\}$. Let $\partial^{=k} f$ denote the set $\{\partial_\mu f : \mu \in \binom{\mathbf{x}}{k}\}$. In other words, $\partial^{=k} f$ is the set of all k-th order partial derivatives of f, while differentiating with respect to a variable at most once. Then we define

$$\mathsf{SPD}_{k,\ell}(f) \stackrel{\text{def}}{=} \dim\left(\mathrm{span}_{\mathbb{F}}\left(\mathbf{x}^{\leq \ell} \cdot \partial^{=k} f\right)\right). \tag{2.1}$$

By a *shift* of a derivative $\partial_\mu f$ we mean an element of $\mathbf{x}^{\leq \ell} \cdot \{\partial_\mu f\}$. A nice property of SPD is that it is subadditive.

Proposition 6 (Subadditivity)**.** *Let $f, g \in \mathbb{F}[\mathbf{x}]$ be two polynomials. Then $\mathsf{SPD}_{k,\ell}(f+g) \leq \mathsf{SPD}_{k,\ell}(f) + \mathsf{SPD}_{k,\ell}(g)$.*

Proof. Let $\mathsf{SPD}_{k,\ell}(f) = u$ and $\mathsf{SPD}_{k,\ell}(g) = v$. Then $U = \mathrm{span}_{\mathbb{F}}(\mathbf{x}^{\leq \ell} \cdot \partial^{=k} f)$ and $V = \mathrm{span}_{\mathbb{F}}(\mathbf{x}^{\leq \ell} \cdot \partial^{=k} g)$ are of dimensions u and v respectively. We have that $\mathsf{SPD}_{k,\ell}(f+g) = \dim(\mathrm{span}_{\mathbb{F}}(\mathbf{x}^{\leq \ell} \cdot \partial^{=k}(f+g)))$ which is at most $u+v$ for the following reason. For any element h from $\mathrm{span}_{\mathbb{F}}(\mathbf{x}^{\leq \ell} \cdot \partial^{=k}(f+g))$ we have

$$\begin{aligned} h &= \sum_{\mu \in \binom{\mathbf{x}}{k}, m \in \mathbf{x}^{\leq \ell}} c_{\mu,m} \cdot \partial_\mu (f+g) \cdot m \qquad (c_{\mu,m}\text{'s are from } \mathbb{F}) \\ &= \sum_{\mu \in \binom{\mathbf{x}}{k}, m \in \mathbf{x}^{\leq \ell}} c_{\mu,m} \cdot \partial_\mu f \cdot m + \sum_{\mu \in \binom{\mathbf{x}}{k}, m \in \mathbf{x}^{\leq \ell}} c_{\mu,m} \cdot \partial_\mu g \cdot m \qquad \begin{array}{l}(\text{from the additivity} \\ \text{of derivatives})\end{array} \\ &\in U + V \end{aligned}$$

where $U + V = \{F + G : F \in U, G \in V\}$ which is of dimension at most $u + v$. □

In the definition of SPD above, the restriction of deriving with respect to a variable at most once is not essential as such, but it eases the presentation in the later chapters as the target polynomial there is multilinear. However, only for the rest of this section let us allow taking derivative with respect to a variable more than once. With this relaxation we can prove the following: SPD of a polynomial remains the same under invertible, linear change of coordinates. (The shifted skewed partials dimension measure used in [KST16b], on the other hand, is not known to have this property.) Precisely, let $|\mathbf{x}| = n$ and let $g_1, \ldots, g_n \in \mathbb{F}[\mathbf{x}]$ be linear forms. For any polynomial $f \in \mathbb{F}[\mathbf{x}]$, let us write $f(\mathbf{g}(\mathbf{x}))$, or simply $f(\mathbf{g})$, to denote the polynomial obtained by applying the substitution $x_i \to g_i(\mathbf{x})$ in f, for $i = 1, \ldots, n$. Then we have

Proposition 7 (Invariance under invertible, linear change of coordinates)**.** *If the substitution* $\mathbf{g}$ *is invertible, then* $\mathsf{SPD}_{k,\ell}(f(\mathbf{g})) = \mathsf{SPD}_{k,\ell}(f(\mathbf{x}))$.

First we show the invariance of partial derivatives' dimension (i.e. without shift). For a set $\mathbf{f}$ of polynomials in variables $\mathbf{x}$, let $\mathbf{f}(\mathbf{g})$ denote the set $\{f(\mathbf{g}) : f \in \mathbf{f}\}$.

Proposition 8. $\operatorname{span}_{\mathbb{F}} \partial^{=k}(f(\mathbf{g})) \subseteq \operatorname{span}_{\mathbb{F}} (\partial^{=k} f)(\mathbf{g})$.

Proof. For brevity, let us just write ∂_i in place of ∂_{x_i}, for $i = 1, \ldots, n$. It suffices to show that every element of $\partial^{=k}(f(\mathbf{g}))$ is a linear combination of elements in $(\partial^{=k} f)(\mathbf{g})$. Particularly, we show (without loss of generality) that the derivative $\partial_{1,\ldots,k}(f(\mathbf{g}))$ equals the sum $\sum_{j_1,\ldots,j_k \in [n]} (\partial_{j_1,j_2,\ldots,j_k} f)(\mathbf{g}) \cdot \prod_{t \in [k]} \partial_t g_{j_t}$. Then, since $g_1, \ldots, g_n$ are linear forms, it follows that $\partial_t g_{j_t}$ is a constant for every $t \in [k]$ and that the sum serves as the required linear combination. It remains to present the proof by induction. For $k = 0$ the claim trivially holds. Assume the inductive hypothesis for $k - 1$. Then

$$
\begin{aligned}
\partial_{1,\ldots,k}(f(\mathbf{g})) &= \partial_1 \partial_{2,\ldots,k}(f(\mathbf{g})) \\
&= \partial_1 \left(\sum_{j_2,\ldots,j_k \in [n]} (\partial_{j_2,\ldots,j_k} f)(\mathbf{g}) \cdot \prod_{t=2,\ldots,k} \partial_t g_{j_t} \right) && \text{(from the inductive hypothesis)} \\
&= \sum_{j_2,\ldots,j_k \in [n]} \partial_1 \left((\partial_{j_2,\ldots,j_k} f)(\mathbf{g}) \right) \cdot \prod_{t=2,\ldots,k} \partial_{i_t} g_{j_t} && \text{(from the linearity of derivatives)} \\
&= \sum_{j_2,\ldots,j_k \in [n]} \left(\sum_{j_1 \in [n]} (\partial_{j_1,j_2,\ldots,j_k} f)(\mathbf{g}) \cdot \partial_1 g_{j_1} \right) \cdot \prod_{t=2,\ldots,k} \partial_t g_{j_t} && \text{(from chain rule)} \\
&= \sum_{j_1,\ldots,j_k \in [n]} (\partial_{j_1,j_2,\ldots,j_k} f)(\mathbf{g}) \cdot \prod_{t \in [k]} \partial_t g_{j_t}.
\end{aligned}
$$

□

We point out that if we stick to the unmodified definition of SPD (where the derivative with respect to a variable can be taken at most once), then the containment $\text{span}_{\mathbb{F}}\ \partial^{=k}(f(\mathbf{g})) \subseteq \text{span}_{\mathbb{F}}\ (\partial^{=k}f)(\mathbf{g})$ does not hold for a general (non-multilinear) f.

Proof of Proposition 7. Our strategy is to show a chain of equalities:

$$\begin{aligned}
\mathsf{SPD}_{k,\ell}(f(\mathbf{g})) &= \dim(\text{span}_{\mathbb{F}}\ (\mathbf{x}^{\leq \ell}) \cdot \text{span}_{\mathbb{F}}\ (\partial^{=k} f(\mathbf{g}))) \\
&= \dim(\text{span}_{\mathbb{F}}\ (\mathbf{x}^{\leq \ell}) \cdot \text{span}_{\mathbb{F}}\ (\partial^{=k} f)(\mathbf{g})) \\
&= \dim(\text{span}_{\mathbb{F}}\ (\mathbf{x}^{\leq \ell}(\mathbf{g})) \cdot \text{span}_{\mathbb{F}}\ (\partial^{=k} f)(\mathbf{g})) \\
&= \dim(\text{span}_{\mathbb{F}}\ (\mathbf{x}^{\leq \ell} \cdot \partial^{=k} f)(\mathbf{g})) \\
&= \mathsf{SPD}_{k,\ell}(f(\mathbf{x})).
\end{aligned}$$

The first (as well as the fourth) equality is easy to verify. The second one comes from Proposition 8 and invertibility of the substitution map $\mathbf{g}$. Invertibility of $\mathbf{g}$ also implies that $\mathbf{x}^{\leq \ell}(\mathbf{g})$ is (ring-)isomorphic to $\mathbf{x}^{\leq \ell}$. Furthermore $\mathbf{x}^{\leq \ell}(\mathbf{g})$ is a subset of $\text{span}_{\mathbb{F}}(\mathbf{x}^{\leq \ell})$ because it only has polynomials of degree at most ℓ, as g_i's are linear. Hence $\text{span}_{\mathbb{F}}\ \mathbf{x}^{\leq \ell} = \text{span}_{\mathbb{F}}\ \mathbf{x}^{\leq \ell}(\mathbf{g})$, justifying the third equality. The last equality is again justified by $\mathbf{g}$'s invertibility. □

Choosing the parameters k and ℓ: Some intuition ([Sah16]). In the rest of this section, we (intuitively) address the following question: For what setting of parameters k, ℓ in $\mathsf{SPD}_{k,\ell}$ can we hope to get the best lower bound? From Section 1.1 it is clear that we want to maximize the ratio $\mathsf{SPD}_{k,\ell}$ of the hard polynomial to $\mathsf{SPD}_{k,\ell}$ of a building block. Let us first find what is the maximum value the $\mathsf{SPD}_{k,\ell}$ of any multilinear polynomial can possibly take.

Let $|\mathbf{x}| = N$ and let $f \in \mathbb{F}[\mathbf{x}]$ be a multilinear polynomial of degree d. From the mapping $\mu \in \binom{\mathbf{x}}{k}$ to $\partial_\mu(f)$ it is clear that $|\partial^{=k} f| \leq \binom{N}{k}$. This, together with the fact that $|\mathbf{x}^{\leq \ell}| = \binom{N+\ell}{N}$, implies $\mathsf{SPD}_{k,\ell}(f) \leq |\mathbf{x}^{\leq \ell} \cdot \partial^{=k} f| \leq \binom{N}{k} \cdot \binom{N+\ell}{N}$. On the other hand, elements of $\partial^{=k} f$ are of degree at most $d-k$ whence that of $\mathbf{x}^{\leq \ell} \cdot \partial^{=k} f$ are of degree at most $d-k+\ell$. Hence $\mathsf{SPD}_{k,\ell}(f) \leq |\mathbf{x}^{\leq \ell} \cdot \partial^{=k} f| \leq \binom{N+d-k+\ell}{N}$. Thus $\mathsf{SPD}_{k,\ell}(f) \leq \min\left(\binom{N}{k} \cdot \binom{N+\ell}{N}, \binom{N+d-k+\ell}{N}\right)$.

To demonstrate the reasoning behind choosing k and ℓ in the particular way we fix them in the later chapters, let us pick the model multi-r-ic depth four formulas with τ-bottom-support. We want to upper bound $\mathsf{SPD}_{k,\ell}$ of a building block or term T in such a circuit. This upper bound has been worked out in Chapter 4 (Lemma 14) and it turns out $\mathsf{SPD}_{k,\ell}(T) \leq \binom{2 \cdot N/\tau + 1}{k} \cdot \binom{N + k \cdot \tau \cdot r + \ell}{N}$.

We are trying to maximize the ratio $\frac{\min\left(\binom{N}{k}\cdot\binom{N+\ell}{N},\binom{N+d-k+\ell}{N}\right)}{\binom{2\cdot N/\tau+1}{k}\cdot\binom{N+k\cdot\tau\cdot r+\ell}{N}}$. In other words, letting $R_1 = \frac{\binom{N}{k}\cdot\binom{N+\ell}{N}}{\binom{2\cdot N/\tau+1}{k}\cdot\binom{N+k\cdot\tau\cdot r+\ell}{N}}$ and $R_2 = \frac{\binom{N+d-k+\ell}{N}}{\binom{2\cdot N/\tau+1}{k}\cdot\binom{N+k\cdot\tau\cdot r+\ell}{N}}$ we want to maximize $\min(R_1, R_2)$. In order for this to make sense we need $R_2 > 1$ which is true only when $d - k > k \cdot \tau \cdot r$. Hence let us say $\epsilon \cdot d \approx k \cdot \tau \cdot r$ where $0 < \epsilon < 1$. This gives us some idea about the value of k in the analysis: we will choose $k \approx \frac{\epsilon \cdot d}{\tau \cdot r}$ for some $0 < \epsilon < 1$. In fact, we will set ϵ to a constant.

With k fixed as above, it remains to set ℓ appropriately. As ℓ increases, we notice that in R_1 the denominator dominates whereas in R_2 the numerator dominates. $\min(R_1, R_2)$ is optimal when $R_1 = R_2$, solving which for ℓ we get

$$\binom{N}{k}\cdot\binom{N+\ell}{N} = \binom{N+d-k+\ell}{N}$$
$$\Rightarrow \binom{N}{k}\cdot\frac{(N+\ell)\ldots(1+\ell)}{N!} = \frac{(N+d-k+\ell)\ldots(1+d-k+\ell)}{N!}$$

$$\begin{aligned}
\Rightarrow \binom{N}{k} &= \frac{(N+d-k+\ell)\ldots(1+d-k+\ell)}{(N+\ell)\ldots(1+\ell)} \\
&= \left(1+\frac{d-k}{N+\ell}\right)\ldots\left(1+\frac{d-k}{1+\ell}\right) \\
&\approx \left(1+\frac{d-k}{1+\ell}\right)^N && \text{(assuming } \ell > N > d-k, \\
& && \text{which will be the case in our analysis.)} \\
&\approx e^{\frac{(d-k)\cdot N}{1+\ell}} \\
\ln\binom{N}{k} &\approx \frac{(d-k)\cdot N}{1+\ell} \\
\ell &\approx \frac{(d-k)\cdot N}{\ln\binom{N}{k}}.
\end{aligned}$$

This is the optimum choice of ℓ we were referring to in Section 1.3. We note that $\ln\binom{N}{k} \leq k \cdot \ln\left(\frac{e\cdot N}{k}\right) < k \cdot \ln(e \cdot N)$. Since $k \approx \frac{\epsilon\cdot d}{r\cdot\tau}$ and since $\tau \geq \log N$ for multi-r-ic depth four formulas with τ-bottom-support in Theorem 2, indeed $\ell \geq \frac{r\cdot N}{\epsilon} > d - k$. Even for multi-$r$-ic depth three formulas, our choice of k would be similar, and the condition $\ell > d - k$ will hold in Chapter 3.

Chapter 3

Depth three Multi-r-ic circuits

In this chapter we prove Theorem 3.

Theorem 3 (Restated). *Let N, d, r be positive integers such that $2^{24} \cdot r^{1.1} \cdot \log N \leq d \leq 0.9 \cdot N$. Then there is an explicit N-variate degree-d multilinear polynomial in VNP such that any multi-r-ic depth three circuit computing it has size $2^{\Omega\left(\frac{d}{r^{1.1} \cdot \log N}\right)}$.*

The proof follows the three-step template described in the previous chapters. We begin with restating the multi-r-ic depth three circuit model.

3.1 Model

Let $\mathbb{F}$ be a field. Let r be a positive integer. A multi-r-ic depth three circuit C computing a polynomial in $\mathbb{F}[\mathbf{x}]$ is of the form

$$C = \sum_{i \in [s]} \prod_{j \in [m_i]} Q_{ij}$$

where $Q_{ij} \in \mathbb{F}[\mathbf{x}]$ are linear polynomials and $\deg_x \left(\prod_{j \in [m_i]} Q_{ij} \right) \leq r$ for every $x \in \mathbf{x}$ and every $i \in [s]$.

Let $|\mathbf{x}| = N$. Our first task is to upper bound $\mathsf{SPD}_{k,\ell}(C)$.

3.2 Upper bounding SPD of a term

For every $i \in [s]$, we call $\prod_{j\in[m_i]} Q_{ij}$ a *term* in C. From subadditivity of SPD (Proposition 6), we have

$$\mathsf{SPD}_{k,\ell}(C) \leq \sum_{i\in[s]} \mathsf{SPD}_{k,\ell}\left(\prod_{j\in[m_i]} Q_{ij}\right). \tag{3.1}$$

Hence it suffices to estimate an upper bound of SPD of a term in C. Let us focus on the i-th term, for some i. For simplicity we drop the subscript "i" and henceforth denote the term with $\prod_{j\in[m]} Q_j$. We call every Q_j a *factor* of the term.

Preprocessing a term: Grouping. Before proceeding with the estimation we preprocess the term as follows. Let τ be a positive integer (to be fixed later). We pick τ many linear factors and multiply them out to get a degree-τ factor. The new factor will replace the old τ factors. We repeat this process until there are less than τ linear factors remaining. We multiply out these remaining factors also into a single factor. Thus at the end, all the factors, except one possibly, have degree τ. Therefore, we assume without loss of generality that for every factor (except possibly one) Q_j,

$$\deg Q_j = \tau. \tag{3.2}$$

Bounding m. Let D denote the degree of the term $\prod_j Q_j$. The model ensures that $D \leq r \cdot N$. This, together with Equation (3.2) which holds for at least $m-1$ factors, implies

$$\begin{aligned}(m-1)\cdot\tau \leq D &\leq r\cdot N\\ \Rightarrow m &\leq \frac{r\cdot N}{\tau} + 1.\end{aligned} \tag{3.3}$$

Lemma 9. *For any $k \leq m$,* $\partial^{=k}\left(\prod_{j\in[m]} Q_j\right) \subseteq \operatorname{span}_{\mathbb{F}}\left(\bigcup_{A\in\binom{[m]}{m-k}} (\mathbf{x}^{\leq k\cdot\tau}\cdot\prod_{j\in A} Q_j)\right)$.

Proof. We induct on k. For $k = 0$ the claim is trivially true. Assume that the inductive

hypothesis is true for $k-1$. Let $\mu \in \binom{\mathbf{x}}{k}$. We need to show that

$$\partial_\mu \left(\prod_{j\in[m]} Q_j \right) \in \operatorname{span}_{\mathbb{F}} \left(\bigcup_{A\in\binom{[m]}{m-k}} (\mathbf{x}^{\leq k\cdot\tau} \cdot \prod_{j\in A} Q_j) \right).$$

Pick some $x \in \mu$. Then

$$\partial_\mu \left(\prod_{j\in[m]} Q_j \right) = \partial_x \partial_{\mu\setminus\{x\}} \left(\prod_{j\in[m]} Q_j \right).$$

From the inductive hypothesis

$$\partial_{\mu\setminus\{x\}} \left(\prod_{j\in[m]} Q_j \right) \in \operatorname{span}_{\mathbb{F}} \left(\bigcup_{B\in\binom{[m]}{m-k+1}} (\mathbf{x}^{\leq (k-1)\cdot\tau} \cdot \prod_{j\in B} Q_j) \right).$$

Therefore

$$\partial_\mu \left(\prod_{j\in[m]} Q_j \right) \in \operatorname{span}_{\mathbb{F}} \left(\bigcup_{B\in\binom{[m]}{m-k+1}} \partial_x(\mathbf{x}^{\leq (k-1)\cdot\tau} \cdot \prod_{j\in B} Q_j) \right). \tag{3.4}$$

Examining $\partial_x(\mathbf{x}^{\leq (k-1)\cdot\tau} \cdot \prod_{j\in B} Q_j)$, an element of it is of the form $\partial_x(q \cdot \prod_{j\in B} Q_j)$ for some $q \in \mathbf{x}^{\leq (k-1)\cdot\tau}$.

$$\begin{aligned}
\partial_x(q \prod_{j\in B} Q_j) = {} & \partial_x q \cdot \prod_{j\in B} Q_j + q \cdot \sum_{j'\in B} \partial_x Q_{j'} \cdot \prod_{j\in B\setminus\{j'\}} Q_j && \text{(by product rule)} \\
= {} & \partial_x q \cdot Q_b \cdot \prod_{j\in B\setminus\{b\}} Q_j + \sum_{j'\in B} q \cdot \partial_x Q_{j'} \cdot \prod_{j\in B\setminus\{j'\}} Q_j && \text{(for any } b \in B\text{)} \\
\in {} & \bigcup_{A\in\binom{B}{m-k}} \operatorname{span}_{\mathbb{F}} \left(\mathbf{x}^{\leq k\cdot\tau} \cdot \prod_{j\in A} Q_j \right)
\end{aligned}$$

where the last step follows from the observation that $\partial_x q, q \in \mathbf{x}^{\leq (k-1)\cdot\tau}$, and $Q_b, \partial_x Q_{j'} \in$

$\mathrm{span}_{\mathbb{F}}(\mathbf{x}^{\leq\tau})$ and in turn $\partial_x q \cdot Q_b,\ q \cdot \partial_x Q_{j'} \in \mathrm{span}_{\mathbb{F}}(\mathbf{x}^{\leq k\cdot\tau})$. Thus we have that

$$\partial_x(\mathbf{x}^{\leq(k-1)\cdot\tau} \cdot \prod_{j\in B} Q_j) \subseteq \bigcup_{A\in\binom{B}{m-k}} \mathrm{span}_{\mathbb{F}}\left(\mathbf{x}^{\leq k\cdot\tau} \cdot \prod_{j\in A} Q_j\right).$$

Plugging this in Equation (3.4) we get

$$\begin{aligned}\partial_\mu\left(\prod_{j\in[m]} Q_j\right) \in\ & \mathrm{span}_{\mathbb{F}}\left(\bigcup_{B\in\binom{[m]}{m-k+1}} \bigcup_{A\in\binom{B}{m-k}} \mathrm{span}_{\mathbb{F}}\left(\mathbf{x}^{\leq k\cdot\tau} \cdot \prod_{j\in A} Q_j\right)\right) \\ =\ & \mathrm{span}_{\mathbb{F}}\left(\bigcup_{A\in\binom{[m]}{m-k}} (\mathbf{x}^{\leq k\cdot\tau} \cdot \prod_{j\in A} Q_j)\right).\end{aligned}$$

□

Lemma 10. *For any $k \leq m$, $\mathsf{SPD}_{k,\ell}\left(\prod_{j\in[m]} Q_j\right) \leq \binom{\frac{r\cdot N}{\tau}+1}{k} \cdot \binom{N+k\cdot\tau+\ell}{N}$.*

Proof. From Lemma 9,

$$\begin{aligned}\partial^{=k}\left(\prod_{j\in[m]} Q_j\right) \cdot \mathbf{x}^{\leq\ell} &\subseteq \mathrm{span}_{\mathbb{F}}\left(\bigcup_{A\in\binom{[m]}{m-k}} (\mathbf{x}^{\leq k\cdot\tau} \cdot \prod_{j\in A} Q_j)\right) \cdot \mathbf{x}^{\leq\ell} \\ &\subseteq \mathrm{span}_{\mathbb{F}}\left(\bigcup_{A\in\binom{[m]}{m-k}} (\mathbf{x}^{\leq k\cdot\tau+\ell} \cdot \prod_{j\in A} Q_j)\right) \\ \Rightarrow \mathsf{SPD}_{k,\ell}\left(\prod_{j\in[m]} Q_j\right) &\leq \binom{m}{k} \cdot |\mathbf{x}^{\leq k\cdot\tau+\ell}| \\ &= \binom{m}{k} \cdot \binom{N+k\cdot\tau+\ell}{N} \\ &\leq \binom{\frac{r\cdot N}{\tau}+1}{k} \cdot \binom{N+k\cdot\tau+\ell}{N} \qquad \text{(from Equation (3.3)).}\end{aligned}$$

□

Corollary 11. *For any $k \leq \frac{r\cdot N}{\tau}+1$, $\mathsf{SPD}_{k,\ell}\left(\prod_{j\in[m]} Q_j\right) \leq \binom{\frac{r\cdot N}{\tau}+1}{k} \cdot \binom{N+k\cdot\tau+\ell}{N}$.*

Proof. The case $k \leq m$ is already handled by Lemma 10. For $m+1 \leq k \leq \frac{r \cdot N}{\tau} + 1$, we observe that $\mathbf{x}^{\leq \ell} \cdot \partial^{=k}(\prod_{j \in [m]} Q_j) \subseteq \operatorname{span}_{\mathbb{F}}\left(\mathbf{x}^{\leq m \cdot \tau + \ell}\right) \subseteq \operatorname{span}_{\mathbb{F}}(x^{\leq k \cdot \tau + \ell})$ (as $m < k$), which suffices. □

3.3 A lower bound on SPD of a hard polynomial

The next step towards proving Theorem 3 is to show an explicit polynomial with large SPD measure. For this purpose we assume the following theorem for this section. A proof is given in Chapter 5.

Theorem 12. *Let N, d be positive integers such that $2^{20} \cdot \log N \leq d \leq 0.9 \cdot N$. Then for any positive integer $k \leq \frac{d}{2^{20} \cdot \log N}$ there exists an explicit N-variate degree-d multilinear homogeneous polynomial $F_{d,k}$ in* VNP *such that*

$$\mathsf{SPD}_{k,\ell}(F_{d,k}) \geq \frac{1}{2} \cdot \binom{N/4001}{k} \cdot \binom{N+\ell}{N} \tag{3.5}$$

where $\ell = \frac{0.006 \cdot d \cdot N}{\ln \binom{N/4001}{k}} - N$.

Remark. One can show that $\ell > 500 \cdot N$ for every choice of d and k allowed by Theorem 12.

3.4 Putting things together

Let the N-variate degree-d polynomial $F(\mathbf{x})$, given by Theorem 12, be the polynomial computed by C. Then $\mathsf{SPD}_{k,\ell}(F) = \mathsf{SPD}_{k,\ell}(C) \leq \sum_{i=1}^{s} \mathsf{SPD}_{k,\ell}\big(\prod_{j \in [t_i]} Q_{ij}\big)$ from Equation (3.1). For any k such that $k \leq \frac{r \cdot N}{\tau}$ and $k \leq \frac{d}{2^{20} \cdot \log N}$, Corollary 11 and Theorem 12 imply

$$\frac{1}{2} \cdot \binom{N/4001}{k} \cdot \binom{N+\ell}{N} \leq \mathsf{SPD}_{k,\ell}(F) = \mathsf{SPD}_{k,\ell}(C) \leq s \cdot \binom{\frac{r \cdot N}{\tau} + 1}{k} \cdot \binom{N + k \cdot \tau + \ell}{N}$$

$$\begin{aligned}
\Rightarrow s &\geq \frac{\frac{1}{2} \cdot \binom{N/4001}{k} \cdot \binom{N+\ell}{N}}{\binom{\frac{r \cdot N}{\tau}+1}{k} \cdot \binom{N+k \cdot \tau+\ell}{N}} \\
&= \frac{1}{2} \cdot \frac{\binom{N/4001}{k} \cdot \frac{(N+\ell)\ldots(1+\ell)}{N!}}{\binom{\frac{r \cdot N}{\tau}+1}{k} \cdot \frac{(N+k \cdot \tau+\ell)\ldots(1+k \cdot \tau+\ell)}{N!}} \\
&= \frac{1}{2} \cdot \frac{\binom{N/4001}{k}}{\binom{\frac{r \cdot N}{\tau}+1}{k} \cdot \frac{(N+k \cdot \tau+\ell)\ldots(1+k \cdot \tau+\ell)}{(N+\ell)\ldots(1+\ell)}} \\
&\geq \frac{1}{2} \cdot \frac{\binom{N/4001}{k}}{\binom{\frac{r \cdot N}{\tau}+1}{k} \cdot \left(1 + \frac{k \cdot \tau}{1+\ell}\right)^N}
\end{aligned}$$

$$
\begin{aligned}
&\geq \frac{1}{2} \cdot \frac{\binom{N/4001}{k}}{\binom{\frac{r\cdot N}{\tau}+1}{k} \cdot e^{\frac{k\cdot\tau}{1+\ell}\cdot N}} && \text{(using exponential upper bound 1 from Section 2.2)} \\
&= \frac{1}{2} \cdot \frac{\binom{N/4001}{k}}{\binom{\frac{r\cdot N}{\tau}+1}{k} \cdot e^{\frac{0.006\cdot d\cdot N}{N+\ell}\cdot\frac{k\cdot\tau}{0.006\cdot d}\cdot\frac{N+\ell}{1+\ell}}} \\
&= \frac{1}{2} \cdot \frac{\binom{N/4001}{k}}{\binom{\frac{r\cdot N}{\tau}+1}{k} \cdot \binom{N/4001}{k}^{\frac{k\cdot\tau}{0.006\cdot d}\cdot\frac{N+\ell}{1+\ell}}} && \left(\text{since } \ell = \frac{0.006\cdot d\cdot N}{\ln\binom{N/4001}{k}} - N\right) \\
&\geq \frac{1}{2} \cdot \frac{\binom{N/4001}{k}}{\binom{\frac{r\cdot N}{\tau}+1}{k} \cdot \binom{N/4001}{k}^{\frac{k\cdot\tau}{0.0055\cdot d}}} && \left(\text{by putting } \frac{N+\ell}{1+\ell} \leq \frac{12}{11}, \text{ as } \ell > 500\cdot N\right) \\
&\geq \frac{1}{2} \cdot \left(\frac{k\cdot\tau}{3\cdot r\cdot N} \cdot \left(\frac{N}{4001\cdot k}\right)^{(1-\frac{k\cdot\tau}{0.0055\cdot d})}\right)^k && \text{(using binomial bounds 3 from Section 2.2)} \\
&= \frac{1}{2} \cdot \left(\frac{k\cdot\tau}{3\cdot r\cdot N} \cdot \left(\frac{N}{4001\cdot k}\right)^{(1-\frac{k\cdot\tau}{\epsilon\cdot d})}\right)^k && (3.6)
\end{aligned}
$$

where $\epsilon = 0.0055$ (say). Finally we set values for k and τ as follows:

$$
\begin{aligned}
k &= \frac{d}{150\cdot\tau\cdot\log N} \quad \text{and} \\
\tau &= 2^{16}\cdot r^{1.1}.
\end{aligned}
$$

Before proceeding we make a few remarks. First, we are omitting floor and ceiling notations for simplicity of exposition. Second, this choice of k and τ meets the requirements of Corollary 11 and Theorem 12: clearly $k < \frac{r\cdot N}{\tau}$ and $k \leq \frac{d}{2^{20}\cdot\log N}$. Also, since it is given that $d \geq 2^{24}\cdot r^{1.1}\cdot\log N$, we have $k = \frac{d}{150\cdot\tau\cdot\log N} \geq \frac{2^{24}\cdot r^{1.1}\cdot\log N}{150\cdot 2^{16}\cdot r^{1.1}\cdot\log N} \geq 1$. Plugging the value for k in Equation (3.6),

$$
\begin{aligned}
s &\geq \frac{1}{2} \cdot \left(\frac{d}{3\cdot r\cdot N\cdot 150\cdot\log N} \cdot \left(\frac{N\cdot 150\cdot\tau\cdot\log N}{4001\cdot d}\right)^{(1-\frac{1}{\epsilon\cdot 150\cdot\log N})}\right)^k \\
&= \frac{1}{2} \cdot \left(\frac{1}{3\cdot r} \cdot \left(\frac{d}{150\cdot N\cdot\log N}\right)^{\frac{1}{\epsilon\cdot 150\cdot\log N}} \cdot \left(\frac{\tau}{4001}\right)^{1-\frac{1}{\epsilon\cdot 150\cdot\log N}}\right)^k \\
&\geq \frac{1}{2} \cdot \left(\frac{1}{3\cdot r} \cdot \left(\frac{1}{N^{1.1}}\right)^{\frac{1}{\epsilon\cdot 150\cdot\log N}} \cdot \left(\frac{\tau}{4001}\right)^{1/1.1}\right)^k \\
&= \frac{1}{2} \cdot \left(\frac{1}{3\cdot r} \cdot 2^{\left(\frac{-1.1}{\epsilon\cdot 150}\right)} \cdot \left(\frac{2^{16}\cdot r^{1.1}}{4001}\right)^{1/1.1}\right)^k
\end{aligned}
$$

$$> \frac{1}{2} \cdot (1.5)^k$$
$$= 2^{\Omega\left(\frac{d}{r^{1.1} \cdot \log N}\right)}.$$

This completes the proof of Theorem 3.

3.5 SPD of a depth three circuit vs. the maximum SPD

In this section we express (the upper bound of) $\mathsf{SPD}_{k,\ell}(C)$, given by Corollary 11 as a fraction of $\binom{N}{k} \cdot \binom{N+\ell}{N} = \mathsf{SPD}^{\max}_{k,\ell}$ (say), the maximum the SPD measure of any polynomial can get (see Section 2.5). This is to aid discussion in Section 1.3, on SPD of elementary symmetric polynomials. Let $r = 1$, $\ell = \frac{\alpha \cdot d \cdot N}{\ln \binom{N}{k}} - N$. Then

$$\frac{\mathsf{SPD}^{\max}_{k,\ell}}{\mathsf{SPD}_{k,\ell}(C)} \geq \frac{\binom{N}{k} \cdot \binom{N+\ell}{N}}{s \cdot \binom{N/\tau+1}{k} \cdot \binom{N+k\cdot\tau+\ell}{N}} \qquad \text{(from Lemma 11)}.$$

Mimicking the calculations in Section 3.4 (till Equation (3.6)) we get

$$\frac{\mathsf{SPD}^{\max}_{k,\ell}}{\mathsf{SPD}_{k,\ell}(C)} \geq \frac{1}{s} \cdot \left(\frac{k \cdot \tau}{3 \cdot N} \cdot \left(\frac{N}{k} \right)^{(1-\frac{12\cdot k\cdot\tau}{11\cdot\alpha\cdot d})} \right)^k.$$

Setting $\tau = \frac{\alpha \cdot d}{12 \cdot k}$, we get

$$\begin{aligned} \frac{\mathsf{SPD}^{\max}_{k,\ell}}{\mathsf{SPD}_{k,\ell}(C)} &\geq \frac{1}{s} \cdot \left(\frac{\alpha \cdot d}{33 \cdot N} \cdot \left(\frac{N}{k} \right)^{1/1.1} \right)^k \\ &\geq \frac{1}{s} \cdot \left(\frac{\alpha \cdot d}{33 \cdot N^{1/11} \cdot k^{1/1.1}} \right)^k. \end{aligned}$$

Chapter 4

Depth four multi-r-ic circuits with low bottom support

This chapter considers the model of depth four multi-r-ic circuits with low bottom support and proves Theorem 2.

Theorem 2 (Restated). *Let N, d, r, τ be positive integers such that $2^{21} \cdot \log N \leq d \leq 0.9 \cdot N$, and $\frac{2^{21}}{5000} \cdot \log N \leq \tau \cdot r \leq \frac{d}{5000}$. Then there is an explicit N-variate degree-d multilinear polynomial in VNP such that any τ-bottom-support multi-r-ic depth four circuit computing it has size at least $\left(\frac{\tau^{20} \cdot d}{N \cdot r}\right)^{\left(\frac{0.00001 \cdot d}{\tau \cdot r}\right)}$.*

The organization and content here are similar to that of Chapter 3.

4.1 Model

Let $\mathbb{F}$ be a field. Let r be a positive integer. A multi-r-ic depth four circuit C computing a polynomial in $\mathbb{F}[\mathbf{x}]$ is of the form

$$C = \sum_{i \in [s]} \prod_{j \in [m_i]} Q_{ij}$$

where $Q_{ij} \in \mathbb{F}[\mathbf{x}]$ are such that $\deg_x \prod_{j \in [m_i]} (Q_{ij}) \leq r$ for every $x \in \mathbf{x}$ and every $i \in [s]$.

In this chapter we further assume that C has τ-bottom-support, i.e. every monomial of Q_{ij} has at most τ variables in it. A consequence of it is that $\deg Q_{ij} \leq \tau \cdot r$.

Let $|\mathbf{x}| = N$ and let the output polynomial degree be d. Our first task is to upper bound $\mathsf{SPD}_{k,\ell}(C)$.

4.2 Upper bounding SPD of a term

For every $i \in [s]$, we call $\prod_{j\in[m_i]} Q_{ij}$ a *term* in C. From Proposition 6 we have

$$\mathsf{SPD}_{k,\ell}(C) \leq \sum_{i\in[s]} \mathsf{SPD}_{k,\ell}\Big(\prod_{j\in[m_i]} Q_{ij}\Big). \tag{4.1}$$

Hence it suffices to upper bound SPD of a term in C. Let us focus on i-th term, for some i. For simplicity we drop the subscript "i" and henceforth denote the term with $\prod_{j\in[m]} Q_j$. We call every Q_j a *factor* of the term.

Preprocessing a term: Grouping. Before proceeding with the estimation we preprocess the term as follows. Suppose there are two factors Q_{j_1}, Q_{j_2} in the term such that each of them has degree less than $\frac{\tau\cdot r}{2}$. Then we multiply out Q_{j_1}, Q_{j_2} to get a (new) single factor (of degree $\deg Q_{j_1} + \deg Q_{j_2}$). The new factor will replace Q_{j_1} and Q_{j_2}. We repeat the process as long as there are two or more factors in the term each having degree less than $\frac{\tau\cdot r}{2}$. When the process ends, all the factors, except one possibly, have degree at least $\frac{\tau\cdot r}{2}$. Moreover all the factors will have degree at most $\tau \cdot r$, since that was initially ensured by the model and nowhere in the preprocessing part have we multiplied out two factors either of which has degree greater than or equal to $\frac{\tau\cdot r}{2}$. In summary we can assume without loss of generality that for every factor (except possibly one) Q_j,

$$\frac{\tau\cdot r}{2} \leq \deg Q_j \leq \tau\cdot r. \tag{4.2}$$

Bounding m. Let D denote the degree of the term $\prod_j Q_j$. The model ensures that $D \leq r\cdot N$. This, together with Equation (4.2) which holds for at least $m-1$ factors, implies

$$\begin{aligned}(m-1)\cdot\frac{\tau\cdot r}{2} \leq \sum_{j\in[m]} \deg Q_j = D \leq r\cdot N\\ m \leq \frac{2\cdot N}{\tau} + 1.\end{aligned} \tag{4.3}$$

Lemma 13. *For any $k \leq m$, $\partial^{=k}\big(\prod_{j\in[m]} Q_j\big) \subseteq \mathrm{span}_{\mathbb{F}}\left(\bigcup_{A\in\binom{[m]}{m-k}} (\mathbf{x}^{\leq k\cdot\tau\cdot r}\cdot\prod_{j\in A} Q_j)\right)$.*

Proof. We induct on k. For $k = 0$ the claim is trivially true. Assume that the inductive hypothesis is true for $k-1$. Let $\mu \in \binom{\mathbf{x}}{k}$. We need to show that

$$\partial_\mu \left(\prod_{j\in[m]} Q_j \right) \in \operatorname{span}_{\mathbb{F}} \left(\bigcup_{A\in\binom{[m]}{m-k}} (\mathbf{x}^{\leq k\cdot\tau\cdot r} \cdot \prod_{j\in A} Q_j) \right).$$

Pick some $x \in \mu$. Then

$$\partial_\mu(\prod_{j\in[m]} Q_j) = \partial_x \partial_{\mu\setminus\{x\}}(\prod_{j\in[m]} Q_j).$$

From the inductive hypothesis

$$\partial_{\mu\setminus\{x\}}(\prod_{j\in[m]} Q_j) \in \operatorname{span}_{\mathbb{F}} \left(\bigcup_{B\in\binom{[m]}{m-k+1}} (\mathbf{x}^{\leq (k-1)\cdot\tau\cdot r} \cdot \prod_{j\in B} Q_j) \right).$$

Therefore

$$\partial_\mu(\prod_{j\in[m]} Q_j) \in \operatorname{span}_{\mathbb{F}} \left(\bigcup_{B\in\binom{[m]}{m-k+1}} \partial_x(\mathbf{x}^{\leq (k-1)\cdot\tau\cdot r} \cdot \prod_{j\in B} Q_j) \right). \tag{4.4}$$

We examine $\partial_x(\mathbf{x}^{\leq (k-1)\cdot\tau\cdot r} \cdot \prod_{j\in B} Q_j)$. An element of it is of the form $\partial_x(q \cdot \prod_{j\in B} Q_j)$ for some $q \in \mathbf{x}^{\leq (k-1)\cdot\tau\cdot r}$.

$$\begin{aligned}
\partial_x(q \prod_{j\in B} Q_j) &= \partial_x q \cdot \prod_{j\in B} Q_j + q \sum_{j'\in B} \partial_x Q_{j'} \cdot \prod_{j\in B\setminus\{j'\}} Q_j \qquad \text{(by product rule)} \\
&= \partial_x q \cdot Q_b \cdot \prod_{j\in B\setminus\{b\}} Q_j + \sum_{j'\in B} q \cdot \partial_x Q_{j'} \cdot \prod_{j\in B\setminus\{j'\}} Q_j \qquad \text{(for any } b \in B) \\
&\in \bigcup_{A\in\binom{B}{m-k}} \operatorname{span}_{\mathbb{F}} \left(\mathbf{x}^{\leq k\cdot\tau\cdot r} \cdot \prod_{j\in A} Q_j \right)
\end{aligned}$$

where the last step follows from the observation that $\partial_x q, q \in \mathbf{x}^{\leq (k-1)\cdot\tau\cdot r}$, and $Q_b, \partial_x Q_{j'} \in$

$\mathrm{span}_{\mathbb{F}}(\mathbf{x}^{\leq \tau \cdot r})$ and in turn $\partial_x q \cdot Q_b,\ q \cdot \partial_x Q_{j'} \in \mathrm{span}_{\mathbb{F}}(\mathbf{x}^{\leq k\cdot\tau\cdot r})$. Thus we have that

$$\partial_x(\mathbf{x}^{\leq (k-1)\cdot\tau\cdot r} \cdot \prod_{j\in B} Q_j) \subseteq \bigcup_{A\in\binom{B}{m-k}} \mathrm{span}_{\mathbb{F}}\left(\mathbf{x}^{\leq k\cdot\tau\cdot r} \cdot \prod_{j\in A} Q_j\right).$$

Plugging this in Equation (4.4) we get

$$\begin{aligned}\partial_\mu\left(\prod_{j\in[m]} Q_j\right) &\in \mathrm{span}_{\mathbb{F}}\left(\bigcup_{B\in\binom{[m]}{m-k+1}} \bigcup_{A\in\binom{B}{m-k}} \mathrm{span}_{\mathbb{F}}\left(\mathbf{x}^{\leq k\cdot\tau\cdot r} \cdot \prod_{j\in A} Q_j\right)\right)\\ &= \mathrm{span}_{\mathbb{F}}\left(\bigcup_{A\in\binom{[m]}{m-k}} (\mathbf{x}^{\leq k\cdot\tau\cdot r} \cdot \prod_{j\in A} Q_j)\right).\end{aligned}$$

□

Lemma 14. *For any $k \leq m$, $\mathsf{SPD}_{k,\ell}(\prod_{j\in[m]} Q_j) \leq \binom{2\cdot N/\tau+1}{k}\binom{N+k\cdot\tau\cdot r+\ell}{N}$.*

Proof. From Lemma 13,

$$\begin{aligned}\partial^{=k}(\prod_{j\in[m]} Q_j)\cdot\mathbf{x}^{\leq\ell} &\subseteq \mathrm{span}_{\mathbb{F}}\left(\bigcup_{A\in\binom{[m]}{m-k}} (\mathbf{x}^{\leq k\cdot\tau\cdot r}\cdot\prod_{j\in A} Q_j)\right)\cdot\mathbf{x}^{\leq\ell}\\ &\subseteq \mathrm{span}_{\mathbb{F}}\left(\bigcup_{A\in\binom{[m]}{m-k}} (\mathbf{x}^{\leq k\cdot\tau\cdot r+\ell}\cdot\prod_{j\in A} Q_j)\right)\\ \Rightarrow \mathsf{SPD}_{k,\ell}(\prod_{j\in[m]} Q_j) &\leq \binom{m}{k}\cdot|\mathbf{x}^{\leq k\cdot\tau\cdot r+\ell}|\\ &= \binom{m}{k}\cdot\binom{N+k\cdot\tau\cdot r+\ell}{N}\\ &\leq \binom{2\cdot N/\tau+1}{k}\cdot\binom{N+k\cdot\tau\cdot r+\ell}{N} \qquad \text{(from Equation (4.3)).}\end{aligned}$$

□

Corollary 15. *For any $k \leq \frac{2\cdot N}{\tau}+1$, $\mathsf{SPD}_{k,\ell}(\prod_{j\in[m]} Q_j) \leq \binom{\frac{2\cdot N}{\tau}+1}{k}\cdot\binom{N+k\cdot\tau\cdot r+\ell}{N}$.*

Proof. The case $k \leq m$ is already handled by Lemma 14. For $m+1 \leq k \leq \frac{2\cdot N}{\tau}+1$, we see that $\partial^{=k}(\prod_{j\in[m]} Q_j)\cdot\mathbf{x}^{\leq\ell} \subseteq \mathrm{span}_{\mathbb{F}}\left(\mathbf{x}^{\leq m\cdot\tau\cdot r+\ell}\right) \subseteq \mathrm{span}_{\mathbb{F}}\left(\mathbf{x}^{\leq k\cdot\tau\cdot r+\ell}\right)$, which suffices. □

4.3 A lower bound on SPD of a hard polynomial

As before, we assume Theorem 12 (a proof is given in Chapter 5).

Theorem 12 (Restated). *Let N, d be positive integers such that $2^{20} \cdot \log N \leq d \leq 0.9 \cdot N$. Then for any positive integer $k \leq \frac{d}{2^{20} \cdot \log N}$ there exists an explicit N-variate degree-d multilinear homogeneous polynomial $F_{d,k}$ in* VNP *such that*

$$\mathsf{SPD}_{k,\ell}(F_{d,k}) \geq \frac{1}{2} \cdot \binom{N/4001}{k} \cdot \binom{N+\ell}{N} \tag{4.5}$$

where $\ell = \frac{0.006 \cdot d \cdot N}{\ln \binom{N/4001}{k}} - N$.

Remark. As mentioned before, $\ell > 500 \cdot N$ for every choice of d and k allowed by Theorem 12.

4.4 Putting things together

Let the N-variate degree-d polynomial $F(\mathbf{x})$, given by Theorem 12, be the polynomial computed by C. Then $\mathsf{SPD}_{k,\ell}(F) = \mathsf{SPD}_{k,\ell}(C) \leq \sum_{i=1}^{s} \mathsf{SPD}_{k,\ell}\left(\prod_{j \in [t_i]} Q_{ij}\right)$ from Equation (4.1). For any k such that $1 \leq k \leq \frac{2 \cdot N}{\tau}$ and $k \leq \frac{d}{2^{20} \cdot \log N}$, Corollary 15 and Theorem 12 imply

$$\frac{1}{2} \cdot \binom{N/4001}{k} \cdot \binom{N+\ell}{N} \leq \mathsf{SPD}_{k,\ell}(F) = \mathsf{SPD}_{k,\ell}(C) \leq s \cdot \binom{2 \cdot N/\tau + 1}{k} \cdot \binom{N + k \cdot \tau \cdot r + \ell}{N}$$

$$\begin{aligned}
\Rightarrow s &\geq \frac{\frac{1}{2} \cdot \binom{N/4001}{k} \cdot \binom{N+\ell}{N}}{\binom{2 \cdot N/\tau + 1}{k} \cdot \binom{N + k \cdot \tau \cdot r + \ell}{N}} \\
&\geq \frac{1}{2} \cdot \frac{\binom{N/4001}{k} \cdot \frac{(N+\ell)\ldots(1+\ell)}{N!}}{\binom{3 \cdot N/\tau}{k} \cdot \frac{(N + k \cdot \tau \cdot r + \ell)\ldots(1 + k \cdot \tau \cdot r + \ell)}{N!}} \\
&= \frac{1}{2} \cdot \frac{\binom{N/4001}{k}}{\binom{3 \cdot N/\tau}{k} \cdot \frac{(N + k \cdot \tau \cdot r + \ell)\ldots(1 + k \cdot \tau \cdot r + \ell)}{(N+\ell)\ldots(1+\ell)}} \\
&\geq \frac{1}{2} \cdot \frac{\binom{N/4001}{k}}{\binom{3 \cdot N/\tau}{k} \cdot \left(1 + \frac{k \cdot \tau \cdot r}{1+\ell}\right)^N} \\
&\geq \frac{1}{2} \cdot \frac{\binom{N/4001}{k}}{\binom{3 \cdot N/\tau}{k} \cdot e^{\frac{k \cdot \tau \cdot r}{1+\ell} \cdot N}} \qquad \text{(using exponential upper bound 1 from Section 2.2)}
\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{2} \cdot \frac{\binom{N/4001}{k}}{\binom{3\cdot N/\tau}{k} \cdot e^{\frac{0.006\cdot d\cdot N}{N+\ell} \cdot \frac{k\cdot\tau\cdot r}{0.006\cdot d} \cdot \frac{N+\ell}{1+\ell}}} \\
&= \frac{1}{2} \cdot \frac{\binom{N/4001}{k}}{\binom{3\cdot N/\tau}{k} \cdot \binom{N/4001}{k}^{\frac{k\cdot\tau\cdot r}{0.006\cdot d} \cdot \frac{N+\ell}{1+\ell}}} \qquad \left(\text{since } \ell = \frac{0.006\cdot d\cdot N}{\ln \binom{N/4001}{k}} - N\right) \\
&\geq \frac{1}{2} \cdot \frac{\binom{N/4001}{k}}{\binom{3\cdot N/\tau}{k} \cdot \binom{N/4001}{k}^{\frac{k\cdot\tau\cdot r}{0.0055\cdot d}}} \qquad \left(\text{by putting } \frac{N+\ell}{1+\ell} \leq \frac{12}{11}, \text{ as } \ell > 500\cdot N\right) \\
&\geq \frac{1}{2} \cdot \left(\frac{k\cdot\tau}{3\cdot e\cdot N} \cdot \left(\frac{N}{4001\cdot k} \right)^{(1-\frac{k\cdot\tau\cdot r}{\epsilon\cdot d})} \right)^{k} \qquad \text{(using binomial bounds 3 from Section 2.2),}
\end{aligned}
\tag{4.6}
$$

where $\epsilon = 0.0055$ (say). We set $k = \frac{\epsilon\cdot d}{21\cdot\tau\cdot r}$. (Again, we are omitting ceil/floor notations for simplicity.) We verify that $1 \leq k \leq \frac{d}{2^{20}\cdot\log N}$ by noting that $\frac{d}{5000} \geq \tau\cdot r \geq \frac{2^{21}}{5000}\cdot\log N$ and $d \geq 2^{21}\cdot\log N$ (given in Theorem 2). That $k \leq \frac{2\cdot N}{\tau}$ is easy to see. Plugging the value of k in Equation (4.6) we get

$$
\begin{aligned}
s &\geq \frac{1}{2} \cdot \left(\frac{\epsilon\cdot d}{3\cdot e\cdot N\cdot 21\cdot r} \cdot \left(\frac{N\cdot 21\cdot\tau\cdot r}{4001\cdot\epsilon\cdot d} \right)^{(1-1/21)} \right)^{\frac{\epsilon\cdot d}{21\cdot\tau\cdot r}} \\
&= \frac{1}{2} \cdot \left(\frac{1}{3\cdot e} \cdot \left(\frac{\epsilon\cdot d}{21\cdot N\cdot r} \right)^{1/21} \cdot \left(\frac{\tau}{4001} \right)^{20/21} \right)^{\frac{\epsilon\cdot d}{21\cdot\tau\cdot r}} \\
&= \frac{1}{2} \cdot \left(\left(\frac{1}{3\cdot e} \right)^{21} \cdot \frac{\epsilon\cdot d}{21\cdot N\cdot r} \cdot \left(\frac{\tau}{4001} \right)^{20} \right)^{\frac{\epsilon\cdot d}{21\cdot 21\cdot\tau\cdot r}} \\
&\geq \left(\frac{\tau^{20}\cdot d}{N\cdot r} \right)^{\frac{0.00001\cdot d}{\tau\cdot r}} .
\end{aligned}
$$

□

Chapter 5

A polynomial family with large SPD

In this chapter we prove Theorem 12 by giving an explicit polynomial family having large SPD complexity.

Theorem 12 (Restated). *Let N, d be positive integers such that $2^{20} \cdot \log N \leq d \leq 0.9 \cdot N$. Then for any positive integer $k \leq \frac{d}{2^{20} \cdot \log N}$ there exists an explicit N-variate degree-d multilinear homogeneous polynomial $F_{d,k}$ in* VNP *such that*

$$\mathsf{SPD}_{k,\ell}(F_{d,k}) \geq \frac{1}{2} \cdot \binom{N/4001}{k} \cdot \binom{N+\ell}{N} \tag{5.1}$$

where $\ell = \frac{0.006 \cdot d \cdot N}{\ln \binom{N/4001}{k}} - N$.

Remarks.

1. One can show that $\ell > 500 \cdot N$ for every choice of d and k allowed by Theorem 12. Also, recall from Section 2.5 the rough estimate of $\frac{(d-k) \cdot N}{\ln \binom{N}{k}}$ for ℓ that maximized $\min(R_1, R_2)$. The estimate almost matches with the value of ℓ of the theorem above.

2. The constant 0.9 above can be changed to any other constant in $(0.5, 1)$. Accordingly the constants $\frac{1}{4001}$ and 0.006 will have to be adjusted.

5.1 SPD and the pairwise monomial distance

Suppose we manage to come up with a polynomial f such that in the set $\partial^{=k} f$, at least $\binom{N/4001}{k}$ derivatives are monomials, all distinct (and hence linearly independent). In this so-far nice scenario we have a promising $\mathsf{SPD}_{k,0} \geq \binom{N/4001}{k}$. However, it is not clear a priori if it scales by a factor of $\frac{1}{2} \cdot \binom{N+\ell}{N}$ (let alone $\binom{N+\ell}{N}$) after applying shifts. The reason is that two

distinct monomials (derivatives) can potentially become equal under multiplication by degree-ℓ monomials (shifts). This prompts us to define *distance* between monomials just as in the previous works (as in [CM14]) involving Nisan-Wigderson polynomial family.

Definition 3. *The distance between two multilinear monomials a and b is defined as the number of variables that appear in a but not in b. (When a and b are of the same degree, the distance is symmetric.)*

Like in previous works, we observe that more the distance between two monomials, fewer the number of common monomials after shifts. Therefore, a large pairwise monomial distance among derivatives may bring about the required high $\mathsf{SPD}_{k,\ell}$. To that effect we have the following lemma and a corollary:

Lemma 16. *Let $|\mathbf{x}| = N$. Let $\mathscr{D}$ be a set of multilinear $\mathbf{x}$-monomials, all of the same degree, such that for every $a, b \in \mathscr{D}$, the distance between a and b is at least $\delta \in \mathbb{N}$. Then for any $\ell \in \mathbb{N}$,*

$$\dim\left(\mathrm{span}_{\mathbb{F}}\{\mathbf{x}^{\leq \ell} \cdot \mathscr{D}\}\right) \geq |\mathscr{D}| \cdot \binom{N+\ell}{N} - \frac{1}{2} \cdot |\mathscr{D}|^2 \cdot \binom{N+\ell-\delta}{N}. \tag{5.2}$$

Corollary 17. *Suppose that there is an N-variate degree-d polynomial f such that $\partial^{=k} f$ is a superset of $\mathscr{D}$. Further suppose that $|\mathscr{D}| = \binom{N/4001}{k}$, $\delta = 0.006 \cdot d$. Then for $\ell = \frac{0.006 \cdot d \cdot N}{\ln \binom{N/4001}{k}} - N$,*

$$\mathsf{SPD}_{k,\ell}(f) \geq \frac{1}{2} \cdot \binom{N/4001}{k} \cdot \binom{N+\ell}{N}.$$

We now prove the lemma and the corollary.

Proof of Lemma 16. Let $\mu_1, \mu_2, \ldots, \mu_{|\mathscr{D}|}$ be the elements of $\mathscr{D}$, each of degree d_0 (say). Then from the inclusion-exclusion principle

$$|\mathbf{x}^{\leq \ell} \cdot \mathscr{D}| \geq \sum_{i=1}^{|\mathscr{D}|} |\mathbf{x}^{\leq \ell} \cdot \mu_i| - \sum_{1 \leq i < j \leq |\mathscr{D}|} |\left(\mathbf{x}^{\leq \ell} \cdot \mu_i\right) \cap \left(\mathbf{x}^{\leq \ell} \cdot \mu_j\right)|. \tag{5.3}$$

Clearly $|\mathbf{x}^{\leq \ell} \cdot \mu_i| = |\mathbf{x}^{\leq \ell}| = \binom{N+\ell}{N}$. Let us estimate an upper bound on the size of the set $\left(\mathbf{x}^{\leq \ell} \cdot \mu_i\right) \cap \left(\mathbf{x}^{\leq \ell} \cdot \mu_j\right) = I_{i,j}$ (say). An observation is that every element of $I_{i,j}$ is divisible by μ_i and μ_j both. In other words every element of $I_{i,j}$ has $LCM(\mu_i, \mu_j)$ as a factor. ($LCM(a, b)$ of two monomials a, b refers to the smallest-degree common multiple of the monomials.) Hence if we pretend to factor out $LCM(\mu_i, \mu_j)$ from each of them, $I_{i,j}$ still has the same number of monomials but each with degree at most $\ell + d_0 - \deg LCM(\mu_i, \mu_j) = \ell_0$ (say). Then $|I_{i,j}|$ is

at most the number of N-variate monomials of degree not exceeding ℓ_0, which is $\binom{N+\ell_0}{N}$. It is given that μ_j has at least δ variables that are not in μ_i. Hence $\deg LCM(\mu_i, \mu_j) \geq d_0 + \delta$. That implies $\ell_0 \leq \ell - \delta$ and in turn $|I_{i,j}| \leq \binom{N+\ell-\delta}{N}$. Plugging the bounds in Equation (5.3) we get

$$|\mathbf{x}^{\leq \ell} \cdot \mathscr{D}| \geq |\mathscr{D}| \cdot \binom{N+\ell}{N} - \frac{|\mathscr{D}| \cdot (|\mathscr{D}| - 1)}{2} \cdot \binom{N+\ell-\delta}{N}.$$

Since $\mathbf{x}^{\leq \ell} \cdot \mathscr{D}$ is a set of monomials, its size equals $\dim\left(\mathrm{span}_{\mathbb{F}}\{\mathbf{x}^{\leq \ell} \cdot \mathscr{D}\}\right)$. Therefore, the inequality above implies the lemma statement. □

Proof of Corollary 17. Rewriting the subtrahend in Equation (5.2),

$$\begin{aligned}
&\frac{1}{2} \cdot |\mathscr{D}|^2 \cdot \binom{N+\ell-\delta}{N} \\
=&\frac{1}{2} \cdot |\mathscr{D}|^2 \cdot \frac{(N+\ell-\delta)\ldots(1+\ell-\delta)}{N!} \\
=&\frac{1}{2} \cdot |\mathscr{D}|^2 \cdot \frac{(N+\ell)\ldots(1+\ell)}{N!} \cdot \frac{(N+\ell-\delta)\ldots(1+\ell-\delta)}{(N+\ell)\ldots(1+\ell)} \\
=&\frac{1}{2} \cdot |\mathscr{D}|^2 \cdot \binom{N+\ell}{N} \cdot \left(1 - \frac{\delta}{N+\ell}\right) \ldots \left(1 - \frac{\delta}{1+\ell}\right) \\
\leq&\frac{1}{2} \cdot |\mathscr{D}|^2 \cdot \binom{N+\ell}{N} \cdot \left(1 - \frac{\delta}{N+\ell}\right)^N \\
\leq&\frac{1}{2} \cdot |\mathscr{D}| \cdot |\mathscr{D}| \cdot \binom{N+\ell}{N} \cdot e^{-\frac{\delta \cdot N}{N+\ell}} \qquad \text{(using exponential upper bound from Section 2.2)} \\
=&\frac{1}{2} \cdot |\mathscr{D}| \cdot e^{\frac{\delta \cdot N}{N+\ell}} \cdot \binom{N+\ell}{N} \cdot e^{-\frac{\delta \cdot N}{N+\ell}} \\
=&\frac{1}{2} \cdot |\mathscr{D}| \cdot \binom{N+\ell}{N},
\end{aligned}$$

where in the penultimate step we used the equality $|\mathscr{D}| = \binom{N/4001}{k} = e^{\frac{\delta \cdot N}{N+\ell}}$ which simply follows from the value of ℓ and δ handed to us. On plugging the values in Equation (5.2), and noting that $\mathsf{SPD}_{k,\ell}(f) \geq \dim\left(\mathrm{span}_{\mathbb{F}}\{\mathbf{x}^{\leq \ell} \cdot \mathscr{D}\}\right)$, the corollary follows. □

The rest of the chapter is devoted to constructing f such that at least $\binom{N/4001}{k}$ of the k-th order derivatives of f are monomials, and the distance between any two of these monomials is at least $\delta = 0.006 \cdot d$.

5.2 Proving Theorem 12

Our job is to come up with f that meets all the conditions of Corollary 17 simultaneously. Assuming for a moment that we already have a set of monomials $\mathscr{D}$ in N variables such that $|\mathscr{D}| = \binom{N/4001}{k}$ and $\delta = 0.006 \cdot d$, how do we form f whose k-th order derivatives include $\mathscr{D}$? We follow a three-step procedure that has been used in previous works, particularly in [KST16a]: First, we preserve a small subset of $\mathbf{x}$-variables, call it $\mathbf{y}$, such that at least $\binom{N/4001}{k}$ many multilinear $\mathbf{y}$-monomials of degree k each can be formed. (This naturally fixes $|\mathbf{y}| = N/4001$.) Then, we work with $\mathbf{x} \setminus \mathbf{y}$ and form a set of monomials $\mathscr{D}$ with the required size $\binom{N/4001}{k}$ and distance δ as per Corollary 17. Finally, we "tag" (i.e. multiply) every monomial of $\mathscr{D}$ with a unique $\mathbf{y}$-monomial and include it as a summand in f. Clearly then, from f one can get back $\mathscr{D}$ by simply differentiating it with respect to every tag. We need to show that the second step is possible, with all details. This we postpone to the next section while merely recording as Lemma 18 here, and using it to complete the proof of Theorem 12.

Definition 4. *For a set S of monomials and any $\mu \in S$, the* index *of μ in S is the number of elements of S that precede μ, under lexicographic ordering.*

Lemma 18. *Let N, d, k be positive integers such that $2^{20} \cdot \log N \leq d \leq 0.9 \cdot N$ and $k \leq \frac{d}{2^{20} \cdot \log N}$. Then there is an explicit set $\mathscr{D}$ of multilinear monomials in $\frac{4000}{4001} \cdot N$ variables of degree $d - k$ such that $|\mathscr{D}| = \binom{N/4001}{k}$, and for any two monomials $a, b \in \mathscr{D}$ the distance between a and b is at least $\delta = 0.006 \cdot d$. Furthermore, for any given monomial, we can determine its membership and (if applicable) index in $\mathscr{D}$ in $\mathsf{poly}(N)$ time.*

Proof of Theorem 12. Let $|\mathbf{x}| = N$ and $\mathbf{y} \subseteq \mathbf{x}$ with $|\mathbf{y}| = N/4001$. Also let $\mathbf{z} = \mathbf{x} \setminus \mathbf{y}$. Readily we have $\mathscr{D}$ in $\mathbf{z}$-variables, imported from Lemma 18 with all the details. Let $\mu_1 \prec \mu_2 \prec \ldots \prec \mu_{|\mathscr{D}|}$ be elements of $\mathscr{D}$, and $\nu_1 \prec \nu_2 \prec \ldots \prec \nu_{\binom{N/4001}{k}}$ be multilinear $\mathbf{y}$-monomials of degree k, under lexicographic ordering. Define

$$F_{d,k}(\mathbf{x}) \stackrel{\text{def}}{=} \sum_{i=1}^{\binom{N/4001}{k}} \mu_i \cdot \nu_i.$$

$F_{d,k}$ is multilinear and homogeneous: every $F_{d,k}$-monomial is multilinear and has degree $\deg \mu_i + \deg \nu_i = d - k + k = d$. To verify that $F_{d,k} \in \mathsf{VNP}$ we use Valiant's coefficient computation criterion. We give a procedure that takes as input a monomial α and outputs its coefficient in $F_{d,k}$ in $\mathsf{poly}(N)$ time. In fact it is enough to check if α is an $F_{d,k}$-monomial or not, since $F_{d,k}$ has 0-1 coefficients. The procedure to check this goes as follows. Split α into β, the $\mathbf{z}$-part, and

γ, the $\mathbf{y}$-part. Check if β is in $\mathscr{D}$ and if yes then find its index h, by invoking the "furthermore" part of Lemma 18. (If $\beta \notin \mathscr{D}$ then α is not an $F_{d,k}$-monomial.) Now that $\beta = \mu_{h+1}$, for α to be an $F_{d,k}$-monomial it is necessary that $\alpha = \mu_{h+1} \cdot \nu_{h+1}$, i.e. $\gamma = \nu_{h+1}$. Check if so is the case by inspecting γ and seeing whether its index in $\{\nu_i : \ i \in [\binom{N/4001}{k}]\}$ is also h —this can be done efficiently. To elaborate, suppose $i_1 < i_2 < \ldots < i_k$ are the subscripts of the $\mathbf{y}$-variables in γ. A (multilinear, degree-k) $\mathbf{y}$-monomial with subscripts $j_1 < j_2 < \ldots < j_k$ precedes γ if and only if the following conditions hold: $j_t < i_t$ for some $t \in [k]$, and $j_u = i_u$ for all $1 \leq u < t$. For a fixed t, while the conditions leave $j_1, \ldots, j_{t-1}$ with no choice but to equal $i_1, \ldots, i_{t-1}$ respectively, they leave j_t with choices $i_{t-1}+1, \ldots, i_t - 1$ and the remaining part (i.e. $j_{t+1}, \ldots, j_k$) with totally $|\binom{\mathbf{y}\setminus[j_t]}{k-t}| = \binom{|\mathbf{y}|-j_t}{k-t}$ choices. This gives a $\mathsf{poly}(N)$-time-computable expression, namely $\sum_{t\in[k]} \sum_{j_t=i_{t-1}+1}^{i_t-1} \binom{|\mathbf{y}|-j_t}{k-t}$, for the index of γ. All the steps above take $\mathsf{poly}(N)$ time each and hence $F_{d,k}$ is in VNP.

Clearly $\partial^{=k} F_{d,k}$ is a superset of $\{\partial_{\nu_i} F_{d,k} : \ i = 1, 2, \ldots, \binom{N/4001}{k}\} = \mathscr{D}$. The values of δ and $|\mathscr{D}|$, supplied by Lemma 18, and the value of ℓ, are precisely what Corollary 17 demands. Hence, applying Corollary 17, $\mathsf{SPD}_{k,\ell}(F_{d,k})$ is at least $\frac{1}{2} \cdot \binom{N/4001}{k} \cdot \binom{N+\ell}{N}$. □

5.3 Monomials with large pairwise distance

Thinking of proving Lemma 18, we first wonder about forming $\mathscr{D}$ by greedily picking $\binom{N/4001}{k}$ monomials with high pairwise distance. The lemma below would be useful in showing that this idea can be made to work. Its proof involves a simple application of probabilistic method and is deferred until Section 5.3.1.

Lemma 19. *Given m variables, it is possible in time $e^{O(m)}$ to generate a set $M(m)$ of $e^{0.001 \cdot m}$ many multilinear monomials that have the following properties: (i) Every monomial of $M(m)$ is of degree $0.9 \cdot \frac{4001}{4000} \cdot m \approx 0.9 \cdot m$, and (ii) the distance between every two distinct monomials a, b of $M(m)$ is at least $0.007 \cdot m$.*

We will use the above lemma to prove Lemma 18.

A remark. Let us think about the "furthermore" part of Lemma 18. If we apply Lemma 19 with $m = \Theta(N)$ then it seems that the best one can do to search $M(\Theta(N))$ for a given monomial is to use brute force, which clearly takes an undesirable $e^{\Theta(N)}$ time. On the other hand, if one were to adapt Lemma 19 in favor of $\mathsf{poly}(N)$-time index-finding of monomials using brute force, then the size of the set $M(\Theta(N))$ would have to be compromised with to $\mathsf{poly}(N)$ (not to mention the corresponding reduction in degree and distance as well). Nevertheless, an

upside of it, in a sense, would be that we would be spending only $\Theta(\log(N))$ out of N variables. This would let us form roughly $\Theta(N/\log N)$ many variable-disjoint sets $M^{(i)}(\Theta(\log N))$, each in $\Theta(\log N)$ variables. [KST16a] showed that a careful recombination of elements of various $M^{(i)}(\Theta(\log N))$'s produces enough monomials with sufficient pairwise distance in such a way to enable efficient determination of membership (and index). The Nisan-Wigderson polynomial family, defined first in [KSS14], is an important ingredient of their technique. We use the same idea as in [KST16a] in the proof of Lemma 18 below:

Proof of Lemma 18. Taking note of the budget of at most $\frac{4000}{4001} \cdot N$ variables and the need to cater for degree $d-k$, we fix an n-sized subset (out of $\frac{4000}{4001} \cdot N$ variables) to work with, where $n = \frac{d-k}{0.9} \cdot \frac{4000}{4001} < \frac{4000}{4001} \cdot N$ as $d \leq 0.9 \cdot N$[1]. We partition the n-sized subset further into $\frac{n}{c \cdot \log n} = n_0$ (say) disjoint subsets of size $c \cdot \log n$ each and call them $Z^{(i)}$, $i \in [n_0]$. Here c is a constant between 1000 and 2000, chosen in such a way that n_0 is a prime number[2]. For every $Z^{(i)}$, let $M^{(i)}(c \cdot \log n)$, or simply $M^{(i)}$, denote the monomial set given by Lemma 19 on $Z^{(i)}$ variables. Since $M^{(i)}$ has at least $e^{0.001 \cdot c \cdot \log n} \geq n \geq n_0$ monomials, we identify $\eta_1^{(i)}, \ldots, \eta_{n_0}^{(i)}$ with the (lexicographically first n_0 many) monomials of $M^{(i)}$. From Lemma 19, these monomials are of degree $0.9 \cdot \frac{4001}{4000} \cdot c \cdot \log n$. Let $\mathbb{K}$ be a prime field of size n_0. Elements of $\mathbb{K}$ will be denoted with $1, 2, 3, \ldots, n_0$. Define

$$\mathscr{D}' \stackrel{\text{def}}{=} \left\{ \prod_{i \in [n_0]} \eta_{h(i)}^{(i)} \right\}_{\substack{h \in \mathbb{K}[t], \\ \deg h = 0.1 \cdot n_0, \\ h \text{ is monic}}}.$$

(The Nisan-Wigderson polynomial is the sum of monomials of $\mathscr{D}'$, and is parametrized by the size of $\mathbb{K}$ and the degree of h.) Clearly the elements of $\mathscr{D}'$ are multilinear and of degree $n_0 \cdot \deg \eta_{h(i)}^{(i)} = 0.9 \cdot \frac{4001}{4000} \cdot n = d - k$ each.

Pairwise monomial distance of $\mathscr{D}'$. For two monomials μ_1 and μ_2 let $\Delta(\mu_1, \mu_2)$ denote the set of variables that appear in μ_1 but not in μ_2. Consider any two monomials $\prod_{i \in [n_0]} \eta_{h(i)}^{(i)}$ and $\prod_{a \in [n_0]} \eta_{g(a)}^{(a)}$ from $\mathscr{D}'$, where $h, g \in \mathbb{K}[t]$ are two different monic polynomials with $\deg h = \deg g = 0.1 \cdot n_0$. Trivially $\Delta(\eta_{h(i)}^{(i)}, \prod_{a \in [n_0]} \eta_{g(a)}^{(a)}) \subseteq \Delta(\eta_{h(i)}^{(i)}, \eta_{g(i)}^{(i)})$. Due to the variable-disjointness of $M^{(i)}$'s, variables appearing in $\eta_{h(i)}^{(i)}$ do not appear in $\eta_{g(a)}^{(a)}$ for every $a \neq i$. Hence $\Delta(\eta_{h(i)}^{(i)}, \eta_{g(i)}^{(i)})$

[1] Without loss of generality n is an integer; see Chapter 2.

[2] From Bertrand-Chebyshev theorem [Erd32], there is always a prime number between α and 2α for $\alpha \geq 1$.

$\subseteq \Delta(\eta^{(i)}_{h(i)}, \prod_{a\in[n_0]} \eta^{(a)}_{g(a)})$ as well. This implies that

$$\Delta(\eta^{(i)}_{h(i)}, \prod_{a\in[n_0]} \eta^{(a)}_{g(a)}) = \Delta(\eta^{(i)}_{h(i)}, \eta^{(i)}_{g(i)}), \tag{5.4}$$

which is of size zero if $h(i) = g(i)$ (trivially) and at least $0.007 \cdot c \cdot \log n$ otherwise, due to Lemma 19.

Now consider $\Delta(\prod_{i\in[n_0]} \eta^{(i)}_{h(i)}, \prod_{a\in[n_0]} \eta^{(a)}_{g(a)})$. Any element of it has to be in $\Delta(\eta^{(i)}_{h(i)}, \prod_{a\in[n_0]} \eta^{(a)}_{g(a)})$ for some $i \in [n_0]$. Also, it can be in $\eta^{(i)}_{h(i)}$ for at most one i, thanks again to the variable-disjointness of $M^{(i)}$'s. Thus $\Delta(\prod_{i\in[n_0]} \eta^{(i)}_{h(i)}, \prod_{a\in[n_0]} \eta^{(a)}_{g(a)}) = \biguplus_{i\in[n_0]} \Delta(\eta^{(i)}_{h(i)}, \prod_{a\in[n_0]} \eta^{(a)}_{g(a)})$. This implies that δ, the distance between $\prod_{i\in[n_0]} \eta^{(i)}_{h(i)}$ and $\prod_{a\in[n_0]} \eta^{(a)}_{g(a)}$, is

$$\begin{aligned}
\delta = |\Delta(\prod_{i\in[n_0]} \eta^{(i)}_{h(i)}, \prod_{a\in[n_0]} \eta^{(a)}_{g(a)})| &= \sum_{i\in[n_0]} |\Delta(\eta^{(i)}_{h(i)}, \eta^{(i)}_{g(i)})| \qquad \text{(from Equation (5.4))} \\
&= \sum_{i\in[n_0],\ h(i)\neq g(i)} |\Delta(\eta^{(i)}_{h(i)}, \eta^{(i)}_{g(i)})| \\
&\geq \sum_{i\in[n_0],\ h(i)\neq g(i)} 0.007 \cdot c \cdot \log n \qquad \text{(from Lemma 19)} \\
&\geq (n_0 - 0.1 \cdot n_0) \cdot 0.007 \cdot c \cdot \log n \qquad (5.5) \\
&\geq 0.006 \cdot n.
\end{aligned}$$

Inequality 5.5 follows from the fact that h and g, being degree-$(0.1 \cdot n_0)$ univariate polynomials, can have the same evaluation on at most $0.1 \cdot n_0$ points from the field. Indeed, this property is at the heart of the design of Nisan-Wigderson polynomial family.

Since $k \leq \frac{d}{2^{20}\cdot \log N}$, we have $n = \frac{d-k}{0.9} \cdot \frac{4000}{4001} \geq \frac{d}{0.9} \cdot \frac{4000}{4001} \cdot (1 - \frac{1}{2^{20}\log N}) \geq d$. Therefore, $\delta \geq 0.006 \cdot d$.

Finding index of a monomial in $\mathscr{D}'$. Let $z^{(i)}_1 \prec \ldots \prec z^{(i)}_{c\cdot\log n}$ denote the elements of $Z^{(i)}$, for $i \in [n_0]$. Also let $z^{(i)}_j \prec z^{(i_0)}_{j_0}$ for every $i \prec i_0$. Recall that $\eta^{(i)}_1 \prec \ldots \prec \eta^{(i)}_{n_0}$.

Let μ be a given monomial in $\uplus_{i\in[n_0]} Z^{(i)}$-variables. Split μ into $\mu_1, \ldots, \mu_{n_0}$, where μ_i is the $Z^{(i)}$-part of μ. Find the index of μ_i in $M^{(i)}$, for all $i \in [n_0]$, by explicitly generating $M^{(i)}$ with the help of Lemma 19 and searching $M^{(i)}$ for μ_i. Let $j_1, \ldots, j_{n_0}$ be the respective indexes. Applying polynomial interpolation on any $0.1 \cdot n_0 + 1$ evaluations from $\{j_1, \ldots, j_{n_0}\}$, find the coefficients of a degree-$(0.1 \cdot n_0)$ polynomial h. Verify that h is monic, and that it evaluates to j_i on the point

i for all n_0 points in $\mathbb{K}$. Success in all the steps above attests μ's membership. Also, the tuple $(j_1, \ldots, j_{n_0})$ gives the index of μ in $\mathscr{D}'$, using a straightforward formula as described in Section 5.2. Every step above takes $\mathsf{poly}(n) = \mathsf{poly}(N)$ time and hence so does the whole procedure.

Size of $\mathscr{D}'$. The nonzero pairwise monomial distance also implies that the size of $\mathscr{D}'$ is exactly $|\{h \in \mathbb{K}[t],\ \deg h = 0.1 \cdot n_0,\ h \text{ is monic}\}|$, which is at least $|\mathbb{K}|^{0.1 \cdot n_0} \geq \left(\frac{n}{c \cdot \log n}\right)^{\frac{0.1 \cdot n}{c \cdot \log n}}$. We aim to show that this is at least $\binom{N/4001}{k}$. It suffices to show $\log \binom{N/4001}{k} \leq \frac{0.1 \cdot n}{c \cdot \log n} \cdot \log \left(\frac{n}{c \cdot \log n}\right)$. Using binomial bounds 3 (Section 2.2) again, we have

$$\log \binom{N/4001}{k} \leq \log \left(\frac{e \cdot N/4001}{k}\right)^k = k \cdot \log \left(\frac{e \cdot N}{4001 \cdot k}\right) \leq k \cdot \left(\log \left(\frac{N}{k}\right) - 10\right), \tag{5.6}$$

while

$$\frac{0.1 \cdot n}{c \cdot \log n} \cdot \log \left(\frac{n}{c \cdot \log n}\right) \geq \frac{0.1 \cdot n}{c \cdot \log n} \cdot \log n^{0.99} \geq \frac{0.099 \cdot n}{c}.$$

The upper bound on k given in Theorem 12 can be rephrased as the condition $d \geq 2^{20} \cdot k \cdot \log N$. Plugging it in the expression for n,

$$n = \frac{d-k}{0.9} \cdot \frac{4000}{4001} \geq k \cdot (2^{20} \cdot \log N - 1) \cdot \frac{1}{0.9} \cdot \frac{4000}{4001},$$

which implies

$$\begin{aligned} \frac{0.099 \cdot n}{c} &\geq k \cdot (2^{20} \cdot \log N - 1) \cdot \frac{1}{0.9} \cdot \frac{4000}{4001} \cdot \frac{0.099}{2000} \\ &> k \cdot (16 \cdot \log N - 16 \cdot 2^{-20}) \end{aligned}$$

which is clearly greater than $k \cdot \left(\log \left(\frac{N}{k}\right) - 10\right)$ of Equation (5.6).

By retaining the (lexicographically) first $\binom{N/4001}{k}$ elements of $\mathscr{D}'$, we get $\mathscr{D}$ with the required properties. □

5.3.1 Proof of Lemma 19

Lemma 19 (Restated). *Given m variables, it is possible in time $e^{O(m)}$ to generate a set $M(m)$ of $e^{0.001 \cdot m}$ many multilinear monomials that have the following properties: (i) Every monomial of $M(m)$ is of degree $0.9 \cdot \frac{4001}{4000} \cdot m \approx 0.9 \cdot m$, and (ii) the distance between every two distinct monomials a, b of $M(m)$ is at least $0.007 \cdot m$.*

Proof. For brevity, let $c_0 = 0.9 \cdot \frac{4001}{4000}$. In Algorithm 1 we outline a greedy way to construct the required monomials. From Algorithm 1, it is clear that the elements of M have the required

Algorithm 1: A greedy algorithm to generate distant monomials

Input : m, an integer
Output: M, a set of monomials in m variables having properties specified in Lemma 19

```
M := ∅
L := array of all multilinear monomials of degree c_0 · m in the lexicographic order
i := 1
while |M| < e^{0.001·m} do
    if the distance between L_i (i.e. the i-th element of L) and ν is at least 0.007 · m for
     all ν ∈ M then
        M := M ∪ {L_i}
    end
    i := i + 1
end
return M
```

degree and distance. The claim below shows that in the *while* loop, the iterator i does not run out of bounds of L as long as $|M| \leq e^{0.001 \cdot m}$.

Claim 20. *Let* $\mathbf{x}$ *be a set of* m *variables. Let* M *be a set of multilinear monomials of degree* $c_0 \cdot m$ *in* $\mathbf{x}$ *(where* $c_0 = 0.9 \cdot \frac{4001}{4000}$*). If* $|M| < e^{0.001 \cdot m}$ *then there always exists an* $\mathbf{x}$*-monomial of degree* $c_0 \cdot m$ *such that the distance between the monomial and* ν *is at least* $0.007 \cdot m$ *for every* $\nu \in M$*.*

Hence M has the sufficient size as well. It remains to estimate the runtime complexity. Initializing L takes time polynomial in $|L| \leq \binom{m}{c_0 \cdot m} \leq 2^m$. In the *while* loop, every increment of i is preluded by checking not more than $|L|$ pairs of monomials for distance. Since computing distance is a linear-time process, at most $m \cdot |L| \leq m \cdot 2^m$ time is spent in an iteration of the *while* loop. Hence the whole *while*-loop takes time at most $m \cdot 2^m \cdot e^{0.001 \cdot m} \leq e^{2 \cdot m}$ in total. □

Proof of Claim 20. We use the probabilistic argument. Consider picking every variable independently with probability $\frac{c_0}{0.99}$ (which is less than 1) and multiplying the picked variables to form a monomial μ (say). Then the expected degree of μ is $E[\deg\ \mu] = \frac{c_0}{0.99} \cdot m$. From Chernoff bound (Section 2.2),

$$\Pr\left[\deg\ \mu < 0.99 \cdot \frac{c_0}{0.99} \cdot m\right] \leq e^{-\frac{0.01^2}{3} \cdot \frac{c_0}{0.99} \cdot m}$$

$$\Rightarrow \Pr\left[\deg\ \mu < c_0 \cdot m\right] < e^{-0.00003 \cdot m} \qquad \text{(by plugging } c_0 = 0.9 \cdot \frac{4001}{4000}\text{)}$$

$$= e_1 \quad \text{(say)}.$$

Recall that $\Delta(a, b)$ denotes the set of variables present in monomial a but not in monomial b. Let $\tilde{\nu}$ be some fixed monomial from M. (Thus $\deg \tilde{\nu} = c_0 \cdot m$.) The distance $|\Delta(\tilde{\nu}, \mu)|$ is then the sum of independent 0-1 random variables $I_1, \ldots, I_{\deg \tilde{\nu}}$, where I_i takes 1 if the i-th variable of $\tilde{\nu}$ is *not* in μ, and takes 0 otherwise. Hence

$$\begin{aligned} E[|\Delta(\tilde{\nu}, \mu)|] &= \sum_{i=1}^{\deg \tilde{\nu}} \Pr[I_i = 1] \qquad \text{(from linearity of expectation)} \\ &= \sum_{i=1}^{c_0 \cdot m} (1 - \frac{c_0}{0.99}) \\ &= c_0 \cdot (1 - \frac{c_0}{0.99}) \cdot m, \end{aligned}$$

and we can apply Chernoff bound again:

$$\begin{aligned} &\Pr[|\Delta(\tilde{\nu}, \mu)| < \ 0.1 \cdot (c_0 \cdot (1 - \frac{c_0}{0.99}) \cdot m)] \leq \ e^{-\frac{0.9^2}{3} \cdot c_0 \cdot (1 - \frac{c_0}{0.99}) \cdot m}. \\ \Rightarrow &\Pr[|\Delta(\tilde{\nu}, \mu)| < \ 0.007 \cdot m] \leq \ e^{-0.022 \cdot m} \quad \text{(by plugging } c_0 = 0.9 \cdot \frac{4001}{4000}). \end{aligned}$$

From union bound, the probability that there is a monomial $\nu \in M$ with $|\Delta(\nu, \mu)| < \ 0.007 \cdot m$ is at most

$$|M| \cdot e^{-0.022 \cdot m} \leq \ e^{0.001 \cdot m} \cdot e^{-0.022 \cdot m} = \ e^{-0.021 \cdot m} = \ e_2 \quad \text{(say)}.$$

Thus, μ has degree at least $c_0 \cdot m$ and distance $|\Delta(\nu, \mu)|$ at least $0.007 \cdot m$ for all $\nu \in M$ with probability at least $1 - e_1 - e_2 = 1 - e^{-0.00003 \cdot m} - e^{-0.021 \cdot m} \gg 0$ (for m large enough). In other words, there exists a monomial μ with distance (from monomials of M) at least $0.007 \cdot m$ and degree at least $c_0 \cdot m$. However we want the degree to be exactly $c_0 \cdot m$. We can chop off a few variables from μ to ensure that. Note that such a chopping does not decrease $|\Delta(\nu, \mu)|$. Moreover, this causes $|\Delta(\mu, \nu)|$ to equal $|\Delta(\nu, \mu)| \geq 0.007 \cdot m$, as desired. □

Chapter 6

Depth four multi-r-ic circuits

In this chapter we want to lift the restriction of low bottom support off multi-r-ic depth four circuits and prove Theorem 1. (Theorem 1 however works with a narrower degree range of $0.51 \cdot N \leq d \leq 0.9 \cdot N$, compared to Theorem 2.) We begin with recording two (not necessarily disjoint) subclasses of depth four multi-r-ic circuits for which the lower bound stated in Theorem 1 is readily established:

1. Consider multi-r-ic depth four circuits that compute F_{d_0,k_0} in N_0 variables (with $0.05 \cdot N_0 \leq d_0 \leq 0.9 \cdot N_0$) and have bottom support bounded by $\tau_0 = 20 \cdot \sqrt{\frac{d_0 \cdot \log d_0}{r}}$. For such a τ_0 and the choice of $k_0 = k_0(d_0, \tau_0, r)$ as in Section 4.4, Theorem 2 gives a lower bound of $N_0^{\left(\frac{0.00001 \cdot d_0}{\tau_0 \cdot r}\right)} = N_0^{\Omega\left(\sqrt{\frac{d_0}{r \cdot \log d_0}}\right)}$, which for $d_0 = \Theta(N_0)$ translates to $2^{\Omega\left(\sqrt{\frac{N_0 \cdot \log N_0}{r}}\right)}$. Note that the choice of τ_0 is consistent with the constraint $\tau_0 \geq \frac{2^{21} \cdot \log N_0}{5000 \cdot r}$ in Theorem 2.

2. Consider multi-r-ic depth four circuits (regardless of which polynomial they compute) that have *sparsity* more than $2^{\sqrt{\frac{N \cdot \log N}{100 \cdot r}}}$ (where N is the number of underlying variables). Sparsity of a depth four circuit refers to the sum of the fanin of nodes at the bottom summation layer. It is trivially a lower bound for circuit size.

Arbitrary as the definitions of the two subclasses may seem, they hint at taking into account the (collective) fanin of the bottom summation layer. Recall that in the proof of Theorem 2 we did not care about sparsity; we managed with lower bounding the top fanin alone. We had the low-bottom-support constraint then. This is not the case anymore, and our proof strategy hinges on reducing general depth four multi-r-ic circuits to the two subclasses. Before elaborating further we describe a new polynomial family H which will eventually be shown to be hard for general depth four multi-r-ic circuits. H is in a sense a "compound" of several instances of F_{d_0,k_0} glued by "auxiliary variables".

6.1 Polynomial H

From here on we just write F_{d_0} to refer to F_{d_0,k_0} with the value of $k_0 = k_0(d_0, \tau_0, r)$ fixed to be the same as that in the proof of Theorem 2 in Section 4.4, with $\tau_0 = 20 \cdot \sqrt{\frac{d_0 \cdot \log d_0}{r}}$. To emphasize that F_{d_0} is in variables from a set S, we may write $F_{d_0}(S)$.

Let $\mathbf{x}, \mathbf{u}, \mathbf{v}$ be sets of variables of sizes $n, n, 0.02 \cdot n$ respectively, making a total of $2.02 \cdot n = N$ (say) variables. We call $\mathbf{u}$-variables and $\mathbf{v}$-variables as *auxiliary variables*. Let d be an integer such that $0.51 \cdot N \leq d \leq 0.9 \cdot N$. Also, we set $d_0 = d - 0.97 \cdot n$. Consider the following N-variate, degree-d polynomial:

$$H(\mathbf{x}, \mathbf{u}, \mathbf{v}) \stackrel{\text{def}}{=} \sum_{S \in \binom{\mathbf{x}}{[0.95 \cdot n, 0.97 \cdot n]}} F_{d_0}(S) \cdot \prod_{i:\ x_i \in S} u_i \cdot \prod_{j=1}^{0.97 \cdot n - |S|} v_j. \tag{6.1}$$

In the analysis below, we would apply Theorem 2 on τ_0-bottom-support multi-r-ic depth four circuits computing $F_{d_0}(S)$. As the theorem has some restrictions on the various parameters, let us verify them first in the following remarks.

Remarks.

1. We call the expression $F_{d_0}(S) \cdot \prod_{i:\ x_i \in S} u_i \cdot \prod_{j=1}^{0.97 \cdot n - |S|} v_j$ as the *summand corresponding to* S. It is easy to see that H is multilinear, homogeneous, and has binary coefficients. Every H-monomial stores information about its summand via the $\mathbf{u}$-part. The $\mathbf{v}$-part enforces homogeneity.

2. To verify that $\deg F_{d_0}(S) \leq 0.9 \cdot |S|$, note that $\deg F_{d_0}(S) = d_0 = d - 0.97 \cdot n \leq 0.9 \cdot N - 0.97 \cdot \frac{N}{2.02} < 0.4199 \cdot N$ whereas $0.9 \cdot |S| \geq 0.9 \cdot (0.95 \cdot n) = 0.9 \cdot \frac{0.95}{2.02} \cdot N > 0.42 \cdot N$.

3. As $d_0 \geq 0.51 \cdot N - \frac{0.97}{2.02} \cdot N > 0.029 \cdot N = \Theta(N)$ and $|S| = \Theta(N)$, clearly the condition $\tau_0 \geq \frac{2^{21} \cdot \log |S|}{5000 \cdot r}$ in Theorem 2 is satisfied. Moreover, since $r \leq (N \cdot \log N)^{0.9}$ (as prescribed by Theorem 1), we have $\tau_0 \cdot r = 20 \cdot \sqrt{r \cdot d_0 \cdot \log d_0} = O\left((d_0 \cdot \log d_0)^{0.95}\right)$, satisfying the other requirement $\tau_0 \cdot r \leq \frac{d_0}{5000}$ in Theorem 2.

Lemma 21. *H defines a polynomial family in* VNP.

Proof. From Valiant's criterion, it is sufficient to give a $\mathsf{poly}(N)$-time algorithm to find the coefficient of any given monomial in H. The coefficient is in fact binary, hence checking whether the given monomial is an H-monomial or not is sufficient. Given a monomial, we simply scan its $\mathbf{u}$-part and find its corresponding S or lack thereof. In case such an S does exist, we check

whether the $\mathbf{x}$-part of the monomial is in $F_{d_0}(S)$, using the procedure described in Section 5.2. We can also easily check if the $\mathbf{v}$-part is consistent with the rest. Thus, in $\mathsf{poly}(N)$ time we can tell if the input monomial is an H-monomial or not. □

The following observation turns out to be quite useful for our proof.

Observation 22. *For every $S \in \binom{\mathbf{x}}{[0.95\cdot n, 0.97\cdot n]}$ there is a 0-1 assignment to the auxiliary variables such that after the assignment H equals $F_{d_0}(S)$.*

Proof. The 0-1 assignment is as follows: Assign v_j 1 if $j \in [0.97\cdot n - |S|]$ and 0 otherwise. Assign u_i 1 if $x_i \in S$ and 0 otherwise. After the assignment to $\mathbf{v}$-variables, only those summands corresponding to $\tilde{S} \in \binom{\mathbf{x}}{[|S|, 0.97\cdot n]}$ survive. Of those, only the summand corresponding to S further survives after the assignment to $\mathbf{u}$-variables. That summand, with $\mathbf{u}$ and $\mathbf{v}$ parts already purged, is simply $F_{d_0}(S)$. □

6.2 Proof of Theorem 1

Recall the two subclasses mentioned in the beginning. The lemma below reveals the proof strategy for Theorem 1. We prove the lemma in the next section.

Lemma 23. *Let H be computed by a depth four multi-r-ic circuit (with parameters d, r as required by Theorem 1). Then either the circuit is in Subclass 2, or there exists an assignment of field values to all auxiliary variables and less than 5% of $\mathbf{x}$-variables that reduces the circuit to a member of Subclass 1 computing F_{d_0}.*

Proof of Theorem 1. $H(\mathbf{x}, \mathbf{u}, \mathbf{v}) \in \mathsf{VNP}$ as defined in the previous section is our target polynomial. If the circuit size is greater than $2^{\sqrt{\frac{N\cdot\log N}{100\cdot r}}}$ (which is the case for circuits in Subclass 2) then there is nothing to prove. Hence we assume otherwise. This lands us in the "or" case of the either-or statement of Lemma 23, where a certain assignment is guaranteed to exist. We employ that assignment and consider the resulting circuit in Subclass 1. Since the act of assigning field values to variables does not increase the circuit size, the forthcoming lower bound analysis remains valid.

Now $F_{d_0}(S)$ is being computed for some $S \in \binom{\mathbf{x}}{[0.95\cdot n, 0.97\cdot n]}$. Hence $N_0 \stackrel{\text{def}}{=} |S|$ and $d_0 = d - 0.97\cdot n$ are the number of variables and the degree respectively at this point. As conditions $d_0 \leq 0.9\cdot N_0$ and $\frac{2^{21}}{5000}\cdot\log N_0 \leq \tau_0\cdot r \leq \frac{d_0}{5000}$ are satisfied (see remarks at the beginning of the previous section), Theorem 2 is applicable on this circuit from Subclass 1. Theorem 2 now implies that the circuit

size (in particular top fanin) should be at least

$$\begin{aligned}
\left(\frac{\tau_0^{20}\cdot d_0}{N_0\cdot r}\right)^{\left(\frac{\epsilon_0\cdot d_0}{\tau_0\cdot r}\right)} &= \left(20^{20}\cdot\left(\frac{d_0\cdot\log d_0}{r}\right)^{10}\cdot\frac{d_0}{N_0\cdot r}\right)^{\left(\frac{\epsilon_0\cdot d_0}{\tau_0\cdot r}\right)} && \text{(where } \epsilon_0 = 0.00001\text{)}\\
&> \left(\left(\frac{d_0\cdot\log d_0}{r}\right)^{10}\cdot\frac{1}{r}\right)^{\left(\frac{\epsilon_0\cdot d_0}{\tau_0\cdot r}\right)} && \text{(as } d_0 \geq 0.029\cdot N \text{ and}\\
&&& \ N_0 \leq \tfrac{0.97}{2.02}\cdot N\text{)}\\
&\geq \left(\left(\frac{0.02\cdot N\cdot\log N}{r}\right)^{10}\cdot\frac{1}{r}\right)^{\left(\frac{\epsilon_0\cdot d_0}{\tau_0\cdot r}\right)}\\
&\geq \left(0.02^{10}\cdot(N\cdot\log N)^{0.1}\right)^{\left(\frac{\epsilon_0\cdot d_0}{\tau_0\cdot r}\right)} && \text{(as } r \leq (N\cdot\log N)^{0.9}\text{)}\\
&\geq N^{\left(0.1\cdot\frac{\epsilon_0\cdot d_0}{20\cdot\sqrt{\frac{d_0\cdot\log d_0}{r}}\cdot r}\right)}\\
&= 2^{\left(\frac{\epsilon_0}{200}\cdot\frac{d_0\cdot\log N}{\sqrt{r\cdot d_0\cdot\log d_0}}\right)}\\
&= 2^{\Omega\left(\sqrt{\frac{N\cdot\log N}{r}}\right)} && \text{(as } d_0 = \Theta(N)\text{).}
\end{aligned}$$

□

6.3 Proof of Lemma 23

If a multi-r-ic depth four circuit C (say) computing H is in Subclass 2 then there is nothing to prove. Hence assume otherwise, namely, that

$$C \text{ has sparsity at most } 2^{\sqrt{\frac{N\cdot\log N}{100\cdot r}}}. \tag{6.2}$$

Under this assumption we now need to show the existence of an assignment of the required kind.

We use the probabilistic method. The sample space of assignments is defined by the following procedure: Independently, retain every $\mathbf{x}$-variable with probability 0.96 and set to 0 with probability 0.04[1]. Let the set of retained variables be denoted with $S \subseteq \mathbf{x}$. If $S \in \binom{\mathbf{x}}{[0.95\cdot n, 0.97\cdot n]}$ then further make assignments to auxiliary variables as described in the proof of Observation 22 to make H equal $F_{d_0}(S)$ after the assignment. Otherwise, assign 0 to all the auxiliary variables.

For any such assignment σ, let C and H with σ applied be respectively denoted by C_σ and H_σ. Since every $\mathbf{x}$-variable is retained or assigned zero independently of others, we can apply

[1]The process is referred to as 'random restriction' in the literature.

Chernoff bound (Section 2.2) to say

$$\Pr_{\sigma}\left[0.95\cdot n \le |S| \le 0.97\cdot n\right] \ge \ 1 - e^{-\frac{1}{3}\cdot(\frac{0.01}{0.96})^2\cdot 0.96\cdot n}.$$

In other words, with probability at most $e^{-\frac{1}{3}\cdot\frac{0.01^2}{0.96}\cdot n}$ the event that $|S| \notin [0.95\cdot n, 0.97\cdot n]$ and in turn its sub-event that $\forall S \in \binom{\mathbf{x}}{[0.95\cdot n, 0.97\cdot n]}$ $H_\sigma \neq F_{d_0}(S)$, occur.

Let us now turn our attention to C_σ. Suppose that most assignments convert C into one with low bottom support. Precisely, suppose that

$$\Pr_{\sigma}\left[C_\sigma \text{ has at least } 20\cdot\sqrt{\frac{d_0\cdot\log d_0}{r}}\text{-bottom-support}\right] \le 2^{-0.01\cdot\sqrt{\frac{N\cdot\log N}{r}}}. \tag{6.3}$$

Then the probability that C_σ is not in Subclass 1 is at most $e^{-\frac{1}{3}\cdot\frac{0.01^2}{0.96}\cdot n} + 2^{-0.01\cdot\sqrt{\frac{N\cdot\log N}{r}}} < 1$, for sufficiently large N. Thus there exists an assignment that converts C into a circuit in Subclass 1 computing $F_{d_0}(S)$ for some $S \subseteq \mathbf{x}$ of size between $0.95\cdot n$ and $0.97\cdot n$. This completes the proof, except that we need yet to show that Inequality (6.3) is indeed true (under Condition (6.2)).

For that purpose, let us examine the effect of assignments on the bottom support. For every monomial μ computed at the bottom multiplication layer, we observe that the assignments substitute 0-1 values for **u**- and **v**-parts of μ. Hence for every μ, only the support of its **x**-part (or "**x**-support" for short) contributes to the bottom support of C_σ. Let μ_σ denote μ with the substitution σ. Clearly even a single **x**-variable of μ being assigned 0 causes μ_σ to equal 0 and not contribute to the bottom support of C_σ. Hence the case of high bottom support for C_σ requires some μ with μ_σ having high **x**-support, which is not likely:

$$\begin{aligned}
&\Pr_{\sigma}\left[\mu_\sigma \text{ has } \mathbf{x}\text{-support at least } 20\cdot\sqrt{\frac{d_0\cdot\log d_0}{r}}\right] \le\ 0.96^{20\cdot\sqrt{\frac{d_0\cdot\log d_0}{r}}} \le\ 2^{-\sqrt{\frac{d_0\cdot\log d_0}{r}}}\\
\Rightarrow\ &\Pr_{\sigma}\left[\text{There exists a } \mu_\sigma \text{ that has } \mathbf{x}\text{-support at least } 20\cdot\sqrt{\frac{d_0\cdot\log d_0}{r}}\right]\\
&\le 2^{-\sqrt{\frac{d_0\cdot\log d_0}{r}}}\cdot 2^{\sqrt{\frac{0.01\cdot N\cdot\log N}{r}}} && \text{(from Condition (6.2) and union bound)}\\
&\le 2^{-\sqrt{\frac{0.02\cdot N\cdot\log N}{r}}}\cdot 2^{\sqrt{\frac{0.01\cdot N\cdot\log N}{r}}} && \text{(as } d_0 \ge 0.029\cdot N\text{)}\\
&\le 2^{-0.01\cdot\sqrt{\frac{N\cdot\log N}{r}}}.
\end{aligned}$$

□

Chapter 7

Conclusion

To summarize, we have improved existing multi-r-ic depth four formula lower bounds and also improved the range of r for which the bound remains superpolynomial. While the proof ideas are along the lines of [KST16b], improvement comes from the choice of a VNP polynomial as the hard polynomial, for high degree range. We have used the original shifted partials (SPD) as the measure. The Skewed Shifted Partials measure, devised by [KST16b] to show lower bounds for low degree range, does not seem to remain superior to the original SPD at high degree range.

Our lower bound for multi-r-ic depth four formulas deteriorates when r exceeds $(N \cdot \log N)^{0.9}$. (The constant 0.9 can be brought arbitrarily close to 1 though, we reiterate.) A significant step would be to prove a lower bound that remains superpolynomial for r much greater than N, say $r = N^2$. This would strengthen our understanding of general depth four multi-r-ic formulas. The depth four model can also be helpful if some of kind of a useful depth reduction from arbitrary depth multi-r-ic formulas to depth four multi-r-ic formulas is discovered. For instance, [RY09] showed a reduction from arbitrary depth multilinear formulas to a special kind of depth four multilinear formulas (called log-product), and used it to simplify the proof of [Raz09]'s result that gave an $n^{\Omega(\log n)}$ multilinear formula lower bound against Det_n and Perm_n.

Our lower bound for multi-r-ic depth three formulas also fails to remain superpolynomial for r beyond $N^{0.9}$. Raising this limit would be a (small) step towards proving general formula lower bounds as well, since at $r = N^{\omega(\sqrt{d})}$ we reach the maximum formal degree of a depth three circuit (obtained from depth-reducing a general $\mathsf{poly}(N)$-sized circuit).

Proving lower bounds for multi-r-ic circuits of higher depth would also be interesting, particularly with $\mathrm{IMM}_{n,d}$ as the hard polynomial. An upper bound of $n^{O(\log d)}$ on multilinear formulas for $\mathrm{IMM}_{n,d}$ is easy to show. (On the other hand Det_n, another important polynomial in VP, is

conjectured to require multilinear formulas of size much larger than $n^{\log n}$.) Hence a matching lower bound (for $r = 1$) would be desirable for $\mathrm{IMM}_{n,d}$. Even with Det_n as the hard polynomial, a lower bound for higher depth multi-r-ic formulas would be substantial progress.

In the multilinear world, a superpolynomial separations between formulas of depth Δ and $\Delta+1$ is known, due to [RY09]. It would be desirable to see a similar separation in the multi-r-ic regime.

[KST16b] generalized the notion of multilinear polynomials by defining *multi-r-ic polynomials*, i.e. polynomials whose individual degree is bounded by r. (Clearly, multi-r-ic circuits compute multi-r-ic polynomials.) One can take a multilinear polynomial, raise some (or all) variables of it to the power r, to get its "multi-r-ic version". A multi-r-ic version, in this sense, of a 'restriction' of IMM was studied by [KST16b] and a better lower bound on multi-r-ic depth four circuits was obtained. In fact the lower bound is $2^{\Omega(\sqrt{N})}$, where N is the number of variables, which does not depend on r at all. It would be interesting to see if a similar result could be obtained for higher depth.

In our definition of multi-r-ic formulas we have *syntactically* bounded the individual degree of polynomials, by introducing the notion of formal degree. It is possible to define a *semantically multi-r-ic* model where one only requires the polynomials computed at intermediate gates to be multi-r-ic. Such a model potentially has high formal degree. Thus, proving nontrivial lower bounds on (syntactically) multi-r-ic formulas may offer some insight on the separation between syntactic and semantic multi-r-ic models. As of now, it is an open question whether semantically multi-r-ic formulas are more powerful than (syntactically) multi-r-ic formulas for a general r. For $r = 1$, they are equally powerful (up to polynomial factor), as shown in [Raz09].

Bibliography

[Agr05] Manindra Agrawal. Proving lower bounds via pseudo-random generators. In *Proceedings of the 25th Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*, pages 92–105, 2005. 3

[AV08] Manindra Agrawal and V. Vinay. Arithmetic circuits: A chasm at depth four. In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 67–75, 2008. 4

[Bür00] Peter Bürgisser. Cook's versus Valiant's hypothesis. *Theoretical Computer Science - Selected papers in honor of Manuel Blum*, 235:71–78, 2000. 3

[CM14] Suryajith Chillara and Partha Mukhopadhyay. Depth-4 Lower Bounds, Determinantal Complexity : A Unified Approach. In *32nd Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 239–250, 2014. 35

[Erd32] Paul Erdös. Beweis eines Satzes von Tschebyschef. *Acta Sci. Math. (Szeged)*, 5:194–198, 1932. 39

[FLMS15] Herv Fournier, Nutan Limaye, Meena Mahajan, and Srikanth Srinivasan. The shifted partial derivative complexity of elementary symmetric polynomials. In *International Symposium on Mathematical Foundations of Computer Science*, pages 324–335. Springer, 2015. 9, 10, 11

[GKKS13] Ankit Gupta, Pritish Kamath, Neeraj Kayal, and Ramprasad Saptharishi. Arithmetic circuits: A chasm at depth three. In *Proceedings of the 54th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 578–587, 2013. 4

[GKKS14] Ankit Gupta, Neeraj Kayal, Pritish Kamath, and Ramprasad Saptharishi. Approaching the chasm at depth four. *Journal of the ACM*, 61(6):33:1–33:16, 2014. iii, 6

[Kay12] Neeraj Kayal. An exponential lower bound for the sum of powers of bounded degree polynomials. In *Electronic Colloquium on Computational Complexity (ECCC)TR12-081*, 2012. 6, 12

[KI04] V. Kabanets and R. Impagliazzo. Derandomizing polynomial identity tests means proving circuit lower bounds. *Computational Complexity*, pages 13(1–2):1–46, 2004. 3

[KLSS] Neeraj Kayal, Nutan Limaye, Chandan Saha, and Srikanth Srinivasan. An Exponential Lower Bound for Homogeneous Depth Four Arithmetic Formulas. In *Proceedings of the 55th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 61–70. iii, 5, 6

[Koi12] Pascal Koiran. Arithmetic circuits: The chasm at depth four gets wider. *Theoretical Computer Science*, 448:56–65, 2012. 4

[KS14] Mrinal Kumar and Shubhangi Saraf. On the power of homogeneous depth 4 arithmetic circuits. In *Proceedings of the 55th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 364–373, 2014. 5, 6

[KS15a] Neeraj Kayal and Chandan Saha. Lower bounds for depth-three arithmetic circuits with small bottom fanin. In *Proceedings of the 30th Annual Computational Complexity Conference (CCC)*, pages 158–208, 2015. 5, 6

[KS15b] Neeraj Kayal and Chandan Saha. Lower bounds for sums of products of low arity polynomials. In *Electronic Colloquium on Computational Complexity (ECCC)TR15-073*, 2015. 5, 6

[KS15c] Neeraj Kayal and Chandan Saha. Multi-k-ic depth three circuit lower bound. In *Proceedings of the 32nd Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 30, pages 527–539, 2015. iii, 5

[KS16] 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)*, pages 35:1–35:29, 2016. 5, 6

[KSS14] Neeraj Kayal, Chandan Saha, and Ramprasad Saptharishi. A super-polynomial lower bound for regular arithmetic formulas. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing (STOC)*, pages 146–153, 2014. iii, 7, 39

[KST16a] 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)*, pages 33:1–33:15, 2016. 5, 7, 37, 39

[KST16b] Neeraj Kayal, Chandan Saha, and Sébastien Tavenas. On the size of homogeneous and of depth four formulas with low individual degree. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing(STOC)*, pages 626–632, 2016. iii, iv, vii, 5, 6, 7, 8, 9, 11, 12, 18, 49, 50

[Nis91] Noam Nisan. Lower bounds for non-commutative computation. In *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing*, pages 410–418, 1991. 3, 6

[NW97] Noam Nisan and Avi Wigderson. Lower bounds on arithmetic circuits via partial derivatives. *Computational Complexity*, 6(3):217–234, 1997. 3, 6

[Raz09] Ran Raz. Multi-linear formulas for permanent and determinant are of super-polynomial size. *Journal of the Association for Computing Machinery*, 56(2), 2009. iii, 4, 49, 50

[Raz10] Ran Raz. Tensor-rank and lower bounds for arithmetic formulas. In *Proceedings of the 42nd Annual ACM Symposium on Theory of Computing (STOC)*, pages 659–666, 2010. iii

[RY08] Ran Raz and Amir Yehudayoff. Balancing syntactically multilinear arithmetic circuits. *Computational Complexity*, 17(4):515–535, 2008. 4

[RY09] Ran Raz and Amir Yehudayoff. Lower Bounds and Separations for Constant Depth Multilinear Circuits. *Computational Complexity*, 18(2):171–207, 2009. iii, iv, 4, 5, 9, 49, 50

[Sah16] Chandan Saha. Shifted partial derivatives. `http://www.cs.tau.ac.il/~shpilka/wact2016/program/slides/Chandan_ShiftedPartials.pptx`, 2016. 19

[SW01] Amir Shpilka and Avi Wigderson. Depth-3 arithmetic circuits over fields of characteristic zero. *Computational Complexity*, 10(1):1–27, 2001. 11

[SY10] 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. 3

[Tav13] Sébastien Tavenas. Improved bounds for reduction to depth 4 and depth 3. *Information and Computation*, pages 240:2–11, 2013. 4

[Val79] L. G. Valiant. Completeness Classes in Algebra. In *Proceedings of the eleventh annual ACM symposium on Theory of computing*, pages 249–261, New York, NY, USA, 1979. ACM Press. 15

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