CHENNAI MATHEMATICAL INSTITUTE

MASTER'S THESIS

Sum of Powers of Univariate Polynomials in Algebraic Complexity Theory

Author:
Abhiroop SANYAL

Supervisor:
Prof. Nitin SAXENA

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science

in the

Department of Computer Science

June 13, 2020

Declaration of Authorship

I, Abhiroop SANYAL, declare that this thesis titled, "Sum of Powers of Univariate Polynomials in Algebraic Complexity Theory" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a Master's degree at Chennai Mathematical Institute.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: Abhiroop Sanyal

Date: 08.06.2020

To the faculty of Chennai Mathematical Institute :

The members of the Faculty appointed to examine the thesis of ABHIROOP SANYAL find it satisfactory and recommend that it be accepted.

_____

Prof. Nitin Saxena, N.Rama Rao Chair, IIT Kanpur

_____

Prof. Partha Mukhopadhyay, Chennai Mathematical Institute

"In mathematics the art of proposing a question must be held of higher value than solving it"

- Georg Cantor, 1867

CHENNAI MATHEMATICAL INSTITUTE

Abstract

Nitin Saxena
Department of Computer Science

Master of Science

Sum of Powers of Univariate Polynomials in Algebraic Complexity Theory

by Abhiroop SANYAL

In this thesis, we study the representation of specific families of univariate polynomials as the ***Sum of Powers*** (SOP) of other univariates :

$$f = \sum_{i=1}^{s} l_i^r$$

We present two sparsity based measures $U_{\mathbb{F}}(\cdot)$ and $S_{\mathbb{F}}(\cdot)$ for this representation and provide upper and lower bounds for specific cases for the families $f_d := (x+1)^d$ and $g_d := \sum_{i=0}^{d} 2^{i^2} x^i$. We also show an unconditional lower bound over *localized integer rings* for both these families.

We also claim that when the families f_d and g_d are represented in the SOP form, then the measures $U_{\mathbb{F}}(\cdot)$ and $S_{\mathbb{F}}(\cdot)$ should be large with respect to this representation (Conjecture 2). We prove that Conjecture 2 implies Valiant's Hypothesis (Conjecture 1) for the family g_d. We also show that if Conjecture 2 is true for the family f_d, then either **CH** $\neq$ **#P**/Poly or **VNP** is exponentially separated from **VP**. This is in spirit of the famous "*derandomization implies hardness*" result of [Kabanets and Impagliazzo, 2003]. Much more improved versions of the results proved in this thesis have been shown in [Dutta, Saxena, and Thierauf, 2020] and [Dutta and Saxena, 2020].

We also study the special case of $r = 2$, i.e the *Sum of Squares* model and show examples of dense candidate polynomials that admit sparse sum of squares representations. We also formulate a few plausible conjectures, Conjecture 3 and Conjecture 4, both of which would imply a stronger version of Conjecture 2, for the measure $S_{\mathbb{F}}(\cdot)$. However, we show a systematic procedure to generate counterexamples to these conjectures and in the process, come up with a few surprising polynomial identities.

Acknowledgements

I would like to begin by thanking Prof. Nitin Saxena for his masterful guidance throughout this project. My first internship in computational complexity during my undergraduate years was with him. That period of two months convinced me that I would like to continue on the path of studying arithmetic circuits and I could think of no one but Nitin to help me tread that path. This work was carried out when he hosted me as a visiting student at IIT Kanpur for a whole year during my Master's studies. Nitin not only helped me develop a crucial understanding of the subject matter but taught me the art of asking the right questions at the right time. If I have become any better at that, all the credit goes to him. I am also extremely grateful to him for offering me the position of a Senior Student Research Associate under his guidance, for the summer of 2019 and hosting me for a week during December 2019.

I would like to thank Prof. Partha Mukhopadhyay at CMI, for introducing me to the wonderful world of computational complexity and for the elegant introduction to the beautiful Schwartz-Zippel Lemma. It was the beauty of that proof which led me to seek an internship under Nitin in order to further explore arithmetic circuits. I would thank Partha for the two courses on classical complexity and pseudorandomness that I took with him. I am grateful to him for offering me the position of teaching assistant for a second course in complexity. He was always enthusiastic and ready to help circumvent any dilemma that I might have had during the five years I spent at CMI.

I would like to thank Pranjal da, my senior at CMI and one of the people I have always looked up to. We worked on the same problem and throughout this journey, he has acted as an informal co-advisor and a loving elder brother. I always battered him with numerous questions, ranging from stupid to outright preposterous and he answered every single one with a calm enthusiasm. I will never forget the academic and non-academic discussions that carried on late into the night. I will always be thankful to him for making this journey so memorable. If possible, someday I'd like to be able to emulate his brilliant ability to tackle hard problems by breaking it down into simpler pieces.

My life at CMI and IIT Kanpur would have been a lot more pale if it hadn't been for all my friends. First and foremost, I would like to thank Subhayan, for being my roommate, lab partner and closest friend during this entire period. Thank you for all the wonderful time we have spent over the past 5 years at CMI and IITK. Thank you for teaching me how to bowl straight and how to swim, although I suppose I let you down on both occassions :(. Love you, man ! Also, I must thank Abhibav, for introducing me to a lot of exciting math and theoretical CS. His cheerful enthusiasm during academic discussions was one of the crucial ingredients that made life in the MTech Lab at IITK so great. I suppose we are going to end up working together very soon and I'm looking forward to that ! Of course, I can never forget Subhasis da, for his wonderful friendship is a treasured possession. Thank you for trying to teach me karate moves, playing billiards and making me feel better by showing me that I'm not the only horrible swimmer at IITK (XD)! I would also like to thank my close friends at CMI : Ritankar, Krishnendu, Srijan, Soham, Arnab and so many others. I apologise for not being able to name you all and like Fermat, I'd like to

shield myself with the excuse that the margin is indeed too small to list the names of all of you wonderful people.

I'd like to thank the administration at CMI and IIT Kanpur for being very supportive and helpful throughout. I thank IITK for sponsoring my visit to WACT, 2019 at ICTS, Bangalore.

There is no possible way in which I could list all the reasons for which I would like to thank my parents, for that list is exponentially larger than the entirety of this thesis itself. I just want to tell them that I love them and hope I've made them proud and will continue to do so. I fondly dedicate this thesis to them. I would also like to tell my grandparents that I love them for pampering me when my parents wouldn't.

Lastly, I'd thank Anushka for putting up with all my eccentricities and for her love and support. Making this possible would have been much harder without you.

Contents

List of Symbols

$\mathbb{Z}$	The ring of integers
$\mathbb{R}$	The field of real numbers
$\mathbb{C}$	The field of complex numbers
R	Any commutative ring
$\mathbb{F}$	Any field
$\overline{\mathbb{F}}$	The algebraic closure of a field $\mathbb{F}$
$\mathrm{GL}_n(\mathbb{C})$	The group of invertible matrices over $\mathbb{C}$
$\mathrm{SL}_n(\mathbb{C})$	The group of invertible matrices over $\mathbb{C}$ with determinant 1
$[m]$	The set $\{0, 1, 2, ..., m\}$

Dedicated to my Parents, for their unwavering support, unconditional love and occasional helpful criticism

Chapter 1

Introduction

1.1 A first look

The central objective of Computer Science is to consider various reasonable models of computation and answer, fundamentally, two sets of questions with respect to a model : what it can achieve and what it cannot. The first set of questions are algorithmic questions while the second set are questions pertaining to lower bounds.

In **Algebraic Complexity Theory**, the chief goal is to understand computation of polynomials, where the basic operations are addition and multiplication. For this, the model of computation that is used are known as ***arithmetic circuits***. Roughly, arithmetic circuits are directed acylic graphs with leaf nodes being either variables or field constants, the internal nodes being + or × gates with each internal node having multiple incoming edges and only one outgoing edge and one output node known as the ***root*** that outputs the polynomial computed by the circuit. The ***size*** of the circuit is defined as the number of edges in it. While studying arithmetic circuits, we are mainly interested in the syntactic computation of polynomials.

The major algorithmic goal of studying arithmetic circuits is to provide an answer to the polynomial identity testing problem which is :

Definition 1. For a circuit family $\mathcal{C}$ of size s computing a polynomial family $f_C(x_1, x_2, ..., x_n)$ of degree d ($C \in \mathcal{C}$), the ***Polynomial Identity Testing (PIT)*** problem asks to find an algorithm, that determines, in time $\text{poly}(s, n, d)$, whether $f_C \equiv 0$ or not ?

However, in this thesis, we will focus mainly on lower bounds, specifically *univariate lower bounds*. What is a lower bound ? A lower bound is simply a statement of the following form : *for a specific polynomial family $\{f_n\}_{n\in\mathbb{N}}$ of degree d, the size of the smallest circuit family C_n computing f_n is $\geq \psi(n, d)$ for some bivariate function ψ*. One can easily show, via a simple counting argument, the following lower bound for "most" circuits, over a finite field $\mathbb{F}$.

Theorem 1. [Folklore] Let $S_{min}(f)$ denote the size of the minimal arithmetic circuit computing f. Then,

$$S_{min}(f) \geq \Omega\left(\sqrt{\binom{n+d}{d}}\right)$$

One can show that the same result as 1 holds for infinite fields as well. For proofs, we refer to [Kayal and Saptharishi, 2014] and [Chen, Kayal, and Wigderson, 2011]. The above number is a lower bound on the number of multiplications in any circuit computing a random polynomial, which was complemented by an upper bound due to [Lovett, 2011], who showed that for any polynomial f of degree d on n variables, $\exists$ a circuit C computing f that has at most $\left(\sqrt{\binom{n+d}{d}}\right) \cdot (nd)^{O(1)}$ many multiplications.

However, the lower bound proofs above are existential i.e they do not point to any specific polynomial family for which the said lower bound holds. The main challenge is to find lower bound statements for *explicit* families of polynomials. The first seminal result in this direction was obtained by [Baur and Strassen, 1983], which still remains the best lower bound for general circuits. This was improved recently, for the case of *Arithmetic Branching Programs* by [Chatterjee et al., 2019].

In this thesis, we will mostly focus on lower bounds for univariate polynomials. We will specifically focus on the Sum of Powers model i.e we will consider specific univariate polynomial families and investigate the representation of these families as sum of powers of univariate polynomials. Measure-based approaches in the lower bound regimes have been considered in [Raz, 2009], [Nisan and Wigderson, 1995], [Gupta et al., 2014] and specifically for univariate polynomials in [Kayal et al., 2015]. We look at two different measures based on the sparsity of polynomials used in the representation. The following is the equational representation of the sum of powers model :

$$f = \sum_{i=1}^{s} l_i^r$$

1.1.1 Contributions of this Thesis

- In Chapter 3, we consider candidate families $f_d := (x+1)^d$ and $g_d := \sum_{i=0}^{d} 2^{i^2} x^i$ and consider two sparsity based measures $S_R(\cdot)$ and $U_R(\cdot)$ for these polynomial families. We show that the *Sum of Powers* model is a *complete* model over fields of zero or large characteristic i.e any polynomial can be computed in this model unless restrictions on the size of the expression (in this case, the *arity* of the sum) are imposed. This is proved in two ways, the first proof is via interpolation while the second, more complicated proof, uses the idea of *sumsets* from additive combinatorics. Following this, we obtain a strong lower bound for a restricted class of representations i.e representation as sum of powers of two polynomials ($s = 2$) and an upper bound for the case of small s . We further state our main conjecture, Conjecture 2, which speaks of establishing a lower bound on the measures $S_{\mathbb{F}}(\cdot)$ and $U_{\mathbb{F}}(\cdot)$ and present an unconditional lower bound in the case where we are looking at representations over *localized*

integer rings. We also show that the measure $S_{\mathbb{F}}$ is *large* for *random* polynomials. These results were partly obtained during private communication of the author with Saxena and Dutta and were published recently in [Dutta, Saxena, and Thierauf, 2020].

- In Chapter 3, we consider the phenomenon of depth-reduction, first established in [Valiant et al., 1983], which allows us to to construct a *shallow depth* circuit computing the same polynomial as the original circuit. We further use the depth-4 reduction technique in [Agrawal and Vinay, 2008] to show that proving Conjecture 2 for the family g_d implies **Valiant's Hypothesis** and for the family f_d we obtain a result akin to the famous derandomization $\implies$ hardness result of Kabanets-Impagliazzo [Kabanets and Impagliazzo, 2003]. This is a special subcase of the stronger result proved in [Dutta, Saxena, and Thierauf, 2020].

- In Chapter 4, we consider the *Sum of Squares* model, a special case of the Sum of Powers model. We investigate a few candidate families and conjectures to solve this special case of Conjecture 2 but show that most of these ideas do not work. However, while proving why they don't work, we come up with a systematic procedure to produce counterexamples and exhibit some striking polynomial identities.

Chapter 2

Preliminaries

2.1 Arithmetic Circuits : Our model of computation

The goal of this thesis is to formulate and understand the implications of certain sparsity measure based conjectures for univariate polynomials, with respect to lower bounds. *Sparsity*, for a univariate polynomial, is simply the number of non-zero monomials in it. We mostly work with modified versions of sparsity, the details of which will be outlined in the following chapter. The model that we work with in this setting is the most natural model of computation for polynomials : *arithmetic circuits*. We define this model below [Shpilka and Yehudayoff, 2010].

Definition 2. *An **arithmetic circuit** is a directed acyclic graph with a unique sink vertex called the **root**. The source vertices are labelled by either formal variables or field constants, and each internal node of the graph is labelled by either + or ×. Nodes compute formal polynomials in the input variables. The edges entering + nodes may also have field constants on them in order to allow the computation of $\mathbb{F}$-linear combinations. The polynomial computed by the circuit is the polynomial computed at the root.*

Without loss of generality, we can assume the circuit to be layered, with edges only between successive layers. If we denote the polynomial computed by a circuit as f, then we shall denote the polynomial computed at any internal node g as f_g. Let us now consider some parameters with respect to a given circuit $\mathcal{C}$ with which we will work frequently.

- The ***size*** of a circuit is the number of nodes and edges in the circuit.
- The ***depth*** of a circuit is given by the length of the longest path from the leaf layer to the root.
- The ***degree*** of any node g is the *total degree* of the polynomial f_g computed at the node (The *total degree* of a polynomial is defined as the degree of the highest degree monomial it contains in a dense representation. For example, the total degree of $x_1^3x_3^2 + x_4^4x_1^2 + x_1^2 + 1$ is 6). We can assume that the root is a + gate, since we do not make any assumptions about the reducibility of the polynomial.
- The ***degree*** of a circuit is the maximal degree of a gate in the circuit. Note that this is the *syntactic degree* of the polynomial computed at the root and may not be equal to the total degree.

2.2 Valiant's Algebraic Complexity Classes

We shall now introduce some of the complexity classes that we would be referring to quite often. These algebraic complexity classes were introduced by Leslie Valiant [Valiant, 1979], [Valiant, 1982] to be the algebraic analogues of the classical complexity classes P and NP (which are defined with respect to Turing Machines as models of computation, as opposed to arithmetic circuits). Let us define the algebraic analogue of P below.

Definition 3. $\mathcal{VP}$ (also called class of p-bounded or p-computable polynomials)

A family of polynomials $\{f_n\}$ over a base field $\mathbb{F}$, is said to be *p-computable* if $\exists$ a polynomial $g : \mathbb{N} \to \mathbb{N}$, such that $\forall\ n$, the number of variables as well as the degree of f_n is upper bounded by $g(n)$ and there is an arithmetic circuit of size at most $g(n)$ computing f_n. The class of all p-computable polynomial over a field $\mathbb{F}$ is known as $\mathbf{VP}_{\mathbb{F}}$ (abbreviation for Valiant's P).

Notice that for a polynomial to be in VP, we need both the degree and the circuit size to be polynomially bounded. Valiant also defined the algebraic analogue of NP, capturing the notion of a "witness" instance by that of a "witness" polynomial. We define this class below.

Definition 4. $\mathcal{VNP}$ (also called class of p-definable polynomials)

A family of polynomials $\{f_n\}$ over a base field $\mathbb{F}$ is said to be *p-definable* if $\exists$ a polynomial family $\{g_n\} \in \mathbf{VP}$, $g_n \in \mathbb{F}\left[x_1, x_2, ..., x_{u(n)}\right]$, such that $\forall\ n$:

$$f_n\left(x_1, x_2...., x_{r(n)}\right) = \sum_{t \in \{0,1\}^{s(n)-r(n)}} g_n\left(x_1, x_2, ..., x_{r(n)}, t_1, t_2, ..., t_{r(n)-s(n)}\right)$$

The class of p-definable polynomials over a field $\mathbb{F}$, is known as $\mathbf{VNP}_{\mathbb{F}}$ (abbreviation for Valiant's NP).

The following inclusion is obvious from the above definitions :

$$\mathbf{VP}_{\mathbb{F}} \subseteq \mathbf{VNP}_{\mathbb{F}}$$

for any field $\mathbb{F}$.

The degree restriction for the polynomials in VP is imposed because the model wishes to capture the notion of efficient evaluation of polynomials at points in the base field. For example, consider a polynomial of exponential degree such as $x^{2^n} y^{2^n}$ over $\mathbb{F}_2$. This can be computed by a polynomial sized arithmetic circuit by repeated squaring but the value of the polynomial at (2, 2) cannot be computed by a polynomial sized Boolean circuit (with standard Boolean operations). Also, the following lemma due to Strassen [Strassen, 1973] shows that we can assume, without loss of generality, that an arithmetic circuit does not compute intermediate polynomials of degree superpolynomial in the size of the output.

Theorem 2 (Homogenization)**.** Let f be an degree d n-variate polynomial computed by a circuit C of size s. Then, for every $0 \leq i \leq d$, there is a homogeneous arithmetic circuit C_i of size at most $O\left(sd^2\right)$, that computes the degree i homogeneous part of f (the sum of all monomials of degree i in f).

A very important family in VP is the *determinant* family, denoted by $\mathbf{DET}_n$. This is the symbolic determinant of an $n \times n$ matrix with the $(i,j) - th$ entry being x_{ij}. Explicitly, this polynomial can be written as :

$$\mathbf{DET}_n = \sum_{\sigma \in S_n} \operatorname{sgn}(\sigma) \prod_{i=1}^{n} x_{i\sigma(i)}$$

The class VNP has been defined to mimic NP in the sense that the $\{0,1\}$-vector t can be thought of as a "witness" and summing over all the witnesses can be thought of as the arithmetic analogue of searching for a witness. Perhaps the most important class of polynomials in VNP is the family of symbolic *permanents*, which we denote as $\mathbf{PER}_n$, which is the permanent of a symbolic $n \times n$ matrix. Formally, the dense representation of the permanent is written in the form :

$$\mathbf{PER}_n = \sum_{\sigma \in S_n} \prod_{i=1}^{n} x_{i\sigma(i)}$$

The permanent function has a very nice combinatorial interpretation : the permanent of the adjacency matrix of an bipartite graph equals the number of perfect matchings on the graph. Another interesting family in VNP is the class of *hamilton cycle polynomials*, denoted by $\mathbf{HC}_n$. This is defined as :

$$\mathbf{HC}_n = \sum_{\sigma} \prod_{i=1}^{n} x_{i\sigma(i)}$$

where the sum is over all cycles $\sigma \in S_n$ of length n. The HC_n evaluated at the adjacency matrix of a digraph equals the number of its hamiltonian cycles.

Just like we have the notion of reduction between algorithms/Turing Machines in classical complexity in the form of Turing or Karp reducibility, we also have the notion of reducibility between polynomials in the form of projections, which we define below.

Definition 5. A polynomial f in m variables (over a base field $\mathbb{F}$) is called a *projection* of a polynomial in n variables g (over a field extension $\mathbb{K}$ of $\mathbb{F}$) ($m \leq n$) , which we denote as $f \leq g$, if f can be written as :

$$f(x_1, x_2, ..., x_m) = g(y_1, y_2, ..., y_n)$$

where $y_i \in \mathbb{F} \cup \{x_1, x_2, ..., x_m\}$.

The following is a natural extension of the above definition to families of polynomials.

Definition 6. A polynomial family $f = \{f_n\}$ is said to be a *p-projection* (short for polynomial projection) of another polynomial family $g = \{g_m\}$ (written as $f \leq_p g$) if $\exists$ a polynomially bounded (on both sides) function $s : \mathbb{N} \to \mathbb{N}$, such that for large enough n_0 :

$$f_n \leq g_{s(n)} \qquad \forall \ \ n \geq n_0$$

The classes VP and VNP are clearly closed under p-projections i.e if f is a polynomial in VP (or VNP), then any p-projection of f is also in VP (or VNP). Now, analogous to the notion of NP-completeness, we define completeness in the arithmetic setting.

Definition 7. A family g in VNP is said to be VNP-*complete* if $f \leq_p g \ \forall$ families $f \in$ VNP.

The following result due to Valiant [Valiant, 1979], [Valiant, 1982] is one of the most central to algebraic complexity.

Theorem 3. The family **HC** of hamiltonian cycle polynomials is VNP-complete. If the base field $\mathbb{F}$ does not have characteristic 2, then the family of symbolic permanents **PER** is also VNP-complete.

Valiant also showed that the family of symbolic determinants is VP-complete with respect to *quasi-polynomial* projections. Precisely stated, the theorem is as follows.

Theorem 4. For any family $\{f_n\}$ in VP, $\exists$ a function $s : \mathbb{N} \to \mathbb{N}$, satisfying $s(n) = n^{O(\log n)}$ such that f_n is a projection of $\mathbf{DET}_{s(n)}$.

The notion of projection is one that captures "hardness" of one problem relative to the other. Stated this way, the family **PER** and **HC** are the "hardest" polynomial families in VNP.

The celebrated conjecture in classical complexity is the P vs. NP problem. An analogue was suggested by Valiant and is now known as ***Valiant's Hypothesis***.

Conjecture 1. $\mathbf{VP} \neq \mathbf{VNP}$.

It is clear that Valiant's Hypothesis is true iff PER is not p-computable. In fact, showing any VNP-complete family is not p-computable proves Valiant's Hypothesis.

2.3 Lower Bounds

One of the major goals of algebraic complexity theorists is to prove lower and upper bounds on the size of circuits computing a given family of polynomials. The state of lower bounds for general circuits has not improved since the following result of Baur and Strassen [Baur and Strassen, 1983].

Theorem 5. Any fan-in 2 circuit that computes the polynomial $f = x_1^{d+1} + x_2^{d+1} + \cdots + x_n^{d+1}$ has size $\Omega(n \log d)$.

A fan-in 2 circuit is one in which each node has indegree 2. However, recently in [Chatterjee et al., 2019] an improvement over this bound was given for a large class of circuits called arithmetic branching programs (ABP). Let us define this model first.

Definition 8. An *algebraic branching program* (ABP) is a layered graph with a unique source vertex (call is s) and a unique sink vertex (call it t). All edges move from layer i to layer $i+1$ and each edge is labelled by a linear polynomial. The following is the polynomial computed by the ABP :

$$f = \sum_{p:s\to t} weight(p)$$

where the sum is over all paths p from the source to the sink and the *weight* of a path is defined as the product of the labels over the edges in p.

The following is the relevant theorem from [Chatterjee et al., 2019]. that improves on the lower bound of Strassen for the special case of ABPs :

Theorem 6. Let $\mathbb{F}$ be a field and let $\text{char}(\mathbb{F}) \neq n$. Then any ABP over $\mathbb{F}$ computing the polynomial $f = \sum_{i=1}^{n} x_i^n$ is of size at least $\Omega\left(n^2\right)$.
Furthermore, when the ABP's edge labels are allowed to be polynomials of degree at most d, then the ABP computing f has size at least $\Omega\left(\frac{n^2}{d}\right)$.

Existing lower bound proofs mostly follow the same template, which we outline below. The outline was first followed by a a seminal result of Kalorkoti [Kalorkoti, 1985]. Note that if the underlying graph in an arithmetic circuit class is a tree, rather than a directed acyclic graph, then such circuits are known as *arithmetic formulas*.

Theorem 7. Any arithmetic formula computing PER_n or DET_n requites $\Omega\left(n^3\right)$ size.

The proof follows the following main steps steps :

1. ***Setup*** : Suppose we have a polynomial $f \in \mathbb{F}[X]$ and a subset of varables $Y \subseteq X$. Write f as $f = \sum_{i=1}^{t} f_i g_i$, where $g_i's$ are monomials in the variable set Y and f_i's are polynomials in the complement set $X \setminus Y$. The transcendence degree of $\{f_1, f_2, ..., f_t\}$ is denoted by $\text{td}_Y(f)$. For a fixed partition $X = \dot{\bigcup}_{i=1}^{r} X_i$, the measure is defined as :

$$\Psi_{[\text{Kal}]} : \mathbb{F}[X] \longrightarrow \mathbb{Z}$$

$$\Psi_{[\text{Kal}]}(f) = \sum_{i=1}^{r} \text{td}_{X_i}(f)$$

2. Show that $\Psi_{[\text{Kal}]}(f)$ measure is *small* for every f that can be computed by a *small* formula.

3. Show that $\Psi_{[\text{Kal}]}(\text{PER}_n)$ and $\Psi_{[\text{Kal}]}(\text{DET}_n)$ is *large*.

Clearly, steps 2 and 3 imply that the *formula-size* of determinant/permanent is large. Here, the notions of *small* and *large* are of course subjective, tailored to the kind of lower bound one might wish to prove. Most major lower bound proofs until now have followed this same template, albeit with different notions of measure [Kayal and Saptharishi, 2014] :

1. ***Basic Decomposition*** : Consider a circuit class $\mathcal{C}$. For every polynomial computed by a circuit $C \in \mathcal{C}$, decompose f as a *small* sum of building blocks.

   For example, in the $\Sigma\Pi\Sigma$ circuits, the building blocks are the *products of linear polynomials* that make up the bottom $\Pi\Sigma$ part.

2. ***Constructing the Complexity Measure*** : Construct a map $\Psi : \mathbb{F}[X] \longrightarrow \mathbb{Z}_{\geq 0}$ which is sub-additive :

   $$\Psi(f_1 + f_2) \leq \Psi(f_1) + \Psi(f_2)$$

   More often than not, Ψ is the rank of a large matrix with entries being linear functions in the coefficients of f. Thus, the sub-additive condition is satisfied for free.

3. ***Potential Usefulness of Measure*** : After constructing the measure Ψ, we need to show that it is *small* on the building blocks. Also, show that $\Psi(f)$ is *large* on a random polynomial f.

4. ***Explicit Lower Bound*** : Find an *explicit* polynomial f for which $\Psi(f)$ is large.

There are a couple of remarks that are to be made here. Firstly, we have not made precise what we mean by an *explicit* polynomial, in step 4. In the case of our problem, which we shall investigate in this thesis, we will show that this requires deep thought and an extensive calculations. So, we shall shed more light on this in the later chapters.

Secondly, one may be curious about the usefulness of showing that the measure is large on a *random* polynomial. While this is not an important step in the proof, this shows the *potential usefulness* of a measure as it means that there are a large number of probable candidate polynomials for which the measure attains a large value and showing any one of them to be explicit would work.

Let us consider a simple example from [Saptharishi, 2020] to demonstrate the above steps. We look at lower bounds on the size of $\Sigma\Pi$ circuits. Suppose $f = \sum_{i=1}^{r} m_i$ where m_i's are monomials. Let Ψ denote the measure that simply counts the number of monomials. Clearly, $\Psi(f) \leq r$ for any f that is computed by a $\Sigma\Pi$ circuit of size s. Now, consider a polynomial $g = (x_1 + x_2 + ... + x_n)^n$. The number of monomials in g is $\binom{2n-1}{n-1}$ which is $n^{\Omega(n)}$ by Stirling's approximation. Thus, this gives a lower bound of $n^{\Omega(n)}$ on $\Sigma\Pi$ circuits computing f.

2.3.1 Some results on lower bounds

The following are some seminal results on lower bounds (other than the one mentioned already) that use this measure-based approach :

1. *The $2^{\Omega(n)}$ lower bound of [Grigoriev and Karpinski, 1998] for **depth-3** circuits computing* DET_n or PER_n *over a finite field* $\mathbb{F}$ *with char*$(\mathbb{F}) \neq 2$: This method considered the matrix $M_k(f)$ of the k-th order partial derivatives of $\mathbb{F}$: the rows are indexed by elements of $\mathbf{F}^{n^2}$, the columns are indexed by the k-th order partial derivatives and the (i,j)-th entry is given by the evaluation of the k-th order partial derivative w.r.t x_j at the point of the row index i. For any set $\mathcal{S} \subseteq \mathbb{F}^{n^2}$, denote the matrix $M_k(S,f)$ to be matrix that takes only those rows from $M_k(f)$ whose indices are in $\mathcal{S}$. The complexity measure was defined to be rank $(M_k(S,f))$.

2. *The $n^{\Omega(\log n)}$ lower bound of [Raz, 2009] for **multilinear** formulas computing* DET_n or PER_n : Consider the polynomial $f \in \mathbb{F}[X]$. For a partition of variables $X = Y \biguplus Z$, let $M_{Y,Z}$ denote the following matrix : rows indexed by monomials in Y, columns indexed by monomials in Z and the $M_{Y,Z}(i,j) = \mathrm{coeff}_f\left(m_i(Y) \cdot m_j(Z)\right)$ where $\mathrm{coeff}_f(m)$ denotes the coefficient of the monomial m in f. The measure was defined as rank $(M_{Y,Z}(f))$.

3. *The $2^{\Omega(\sqrt{n})}$ lower bound of [Gupta et al., 2014] for any **depth-4** $\Sigma\Pi^{O(\sqrt{n})}\Sigma\Pi^{\sqrt{n}}$ circuit computing* DET_n or PER_n : Denote by $\partial^{=k}(f)$, the set of al k-th order partial derivatives of f and $\mathbf{x}^{\leq l}$, the set of all monomials of degree at most l. The shifted partials of f, denoted by $\left(\langle \partial^{=k}(f)\rangle_{\leq l}\right)$, is the vector space V spanned by $\{\mathbf{x}^{\leq l} \cdot \partial^{=k}(f)\}$. The measure is defined as dim (V).

2.3.2 Univariate Lower bounds

[Bürgisser, Clausen, and Shokrollahi, 1997] showed that for explicit polynomials $\sum_{i=0}^{d} \sqrt{p_i} x^i$ requires circuits of size $\Omega\left(\sqrt{d/\log d}\right)$, where p_i is the i-th prime. [Strassen, 1974] showed that for integral coefficients, the polynomial $\sum_{i=0}^{d} 2^{2^i} x^i$ requires circuits of size $\Omega\left(\sqrt{d/\log d}\right)$. However, while these polynomials can be converted to *exponentially hard* multilinear polynomials, which is what we need for a proof of Theorem 19, they fail to be *explicit* (in what exact sense, will be pointed out in Chapter 4). Thus, we consider different polynomial families in our study.

[Koiran, 2011] implicitly showed that if there exists a univariate polynomial family $f_d(x)$ of degree d, such that any representation of the form $f_d(x) = \sum_{i=1}^{s} c_i p_i^{e_i}$ where sparsity of $p_i \leq t$ and arbitrary e_i's requires $s \geq (d/t)^{\Omega(1)}$, then **VP** $\neq$ **VNP**. In the case, where $\deg(p_i) \leq t$, a lower bound of $\Omega\left(\sqrt{d/t}\right)$ is known due to [Kayal et al., 2015] (where one proof they give uses the shifted partial derivative approach mentioned earlier). However, notice that this is not even close to the original hypothesis, where t is the sparsity (Consider a polynomial of the form $x^d + 1$. It has degree d but sparsity only 2. Thus, the asymptotic sparation between degree and sparsity could be infinite for a polynomial family). For $\deg(p_i) \leq 1$, $\Omega(d)$ lower bounds are known for certain special families [Garca-Marco and Koiran, 2017].

2.4 Defining the Sparsity-based Measures

2.4.1 The *Support-Union* Measure

We recall the *Sum of Powers* model representation over a univariate polynomial ring $R[x]$:

$$f = \sum_{i=1}^{s} c_i l_i^r \tag{2.1}$$

In the context of arithmetic circuits, note that s is the top fan-in when f is considered to be a circuit. For a fixed representation f of the above form, $\left|\bigcup_{i=1}^{s} \text{supp}\,(l_i)\right|$, represents the number of distinct monomials used in the above representation. We define the *support-union* measure, denoted by $U_R(f,r,s)$, as follows :

$$U_R(f,r,s) := \begin{cases} \min_{f=\sum_{i=1}^{s} c_i l_i^r} \left|\bigcup_{i=1}^{s} \text{supp}\,(l_i)\right| & \text{if such a representation exists for fixed } r \text{ and } s \\ \infty & \text{otherwise} \end{cases}$$

Clearly, this measure is subadditive. Also, over fields, note that $U_{\overline{\mathbb{F}}}(f,r,s) \leq U_{\mathbb{F}}(f,r,s)$, since a representation over a field extension might allow a smaller representation because of possible larger number of cancellations.

2.4.2 The *Sparsity-Sum* Measure

We define a second, coarser measure for the SOP representation. The *Sparsity-Sum* measure for the SOP representation of a polynomial f over a ring R, denoted by $S_R(f,r,s)$, is defined as follows :

$$S_R(f,r,s) := \begin{cases} \min_{f=\sum_{i=1}^{s} c_i l_i^r} \left(\sum_{i=1}^{s} |l_i|_1\right) & \text{if such a representation exists for fixed } r \text{ and } s \\ \infty & \text{otherwise} \end{cases}$$

Here, $|l_i|_1$ denotes the sparsity of the polynomial l_1 and hence, the name of the measure. Notice that $S_R(\cdot) \geq U_R(\cdot)$, for any polynomial f and fixed r,s, over any ring R.

2.5 Depth-Reduction

In Chapter 3, we use depth reduction results to establish a striking connection between Conjecture 2 and Valiant's Hypothesis. Depth-reduction of any given *formula* to an equivalent (i.e computing the same polynomial) formula of depth logarithmic in the size of the original formula was obtained by [Brent, 1974]. Later on, in a landmark paper, Valiant, Skyum, Berkowitz and Rackoff [Valiant et al., 1983] showed that any arithmetic circuit of size s computing a polynomial of degree d in n-variables can be converted into an equivalent circuit of depth $O(\log d)$ and size $\text{poly}(s,n,d)$ i.e only at a cost of polynomial blow up in size. This will be an important starting point for our proof of Theorem 19.
A further striking result was obtained by [Agrawal and Vinay, 2008] and further improved by [Koiran, 2012] and [Tavenas, 2015] who showed a reduction to circuit of *constant* depth (depth-4) with only a super polynomial blow up in size. The proof

technique used here will be crucial in our proof of Theorem 19.
The current state of the art in depth reduction is the result obtained [Gupta et al., 2016] who showed that any n-variate degree d polynomial over rationals, that can be computed by a circuit of size s can also be computed by an equivalent depth-3 circuit of size $s^{O\left(\sqrt{d}\right)}$.

2.5.1 Some useful results

We shall use the following inequality involving binomials in our work :

Lemma 1. For $n \geq k > 0$, we have :

$$\sum_{i=0}^{k} \binom{n}{k} \leq \left(\frac{en}{k}\right)^k$$

For sake of completion, we include a proof.

Proof. For $x = \frac{k}{n}$:

$$\sum_{i=0}^{k} \binom{n}{i} \leq \frac{1}{x^k} \sum_{i=0}^{k} \binom{n}{i} x^i \leq \frac{(1+x)^n}{x^k}$$

However, $(1+x) \leq e^x \ \forall \ x \neq 0$. This gives us :

$$\sum_{i=0}^{k} \binom{n}{i} \leq \frac{e^{xn}}{x^k}$$

Plugging in the value of x in the R.H.S gives us the desired bound. □

We will also use the following result for asymptotic analysis.

Lemma 2. [Bertrand's Postulate] For any $n \in \mathbb{N}$, there exists a prime in the interval $[n, 2n]$.

Chapter 3

The Sparsity Measures and Unconditional Lower Bounds

In the previous chapter, we mostly spoke of polynomials over a field. However, in this chapter, we will mostly adopt the more general setting of polynomials over a ring R. Consider a univariate polynomial $f \in R[x]$ and a positive integer r. We say that f is a sum of r-th powers of polynomials if $\exists$ constants $c_i \in R$, such that f admits the following representation, which we shall call the ***Sum of Powers*** (SOP) model :

$$f = \sum_i c_i l_i^r$$

for some $l_i \in R[x]$. Firstly, we note that this representation is not unique. Consider the polynomial expression $x^4 + 4x^2 = \left(\frac{1}{\sqrt{2}}\left(x^2+2x\right)\right)^2 + \left(\frac{1}{\sqrt{2}}\left(x^2-2x\right)\right)^2$, which is the sum of two squares written in two different ways.

We wish to investigate *two* measures on the SOP model in this thesis : the *support-union* measure and the *sparsity-sum* measure. We will mostly focus on two candidate polynomial families $f_d := (x+1)^d$ and $g_d := \sum_{i=1}^{d} 2^{i^2} x^i$. These two polynomials have nice behavior in the sense that they are *explicit* under no/mild standard complexity theoretic assumptions.

In this chapter, we'll show that the SOP model is complete for a fixed r, using the two different proofs. The first representation, using interpolation, will be crucial in the proof of an upper bound in this chapter (Theorem 10) and when we establish the connection between Conjecture 2 and Valiant's Hypothesis. The second proof brings the idea of *sumsets* into play which we will revisit in Chapter 5, where we study the special case of *Sum of Squares* representation.

We present Conjecture 2, which claims that if we represent specific explicit families of polynomials as sum of *slow -growing* powers of polynomials, then the measures that we define must be large with respect to such a representation. We further prove an unconditional lower bound for our two measures over any *localized integer ring*.

3.1 Completeness of the *Sum of Powers* model

For a fixed r, sum of r-th powers of polynomials is a *complete model* in the case $R = \mathbb{F}$, where $\mathbb{F}$ has 0 or large characteristic [Dutta, Saxena, and Thierauf, 2020]. By a *complete model*, we mean that any polynomial can be computed in this model, without

any restrictions on size of the expression (in this case, the *arity* of the sum). For a simple example, notice that since every polynomial is a sum of monomials the $\Sigma\Pi$ representation is trivially complete. We present the proof , by explicitly showing how to represent a polynomial in the SOP (sum of powers) model, in two different ways. The first proof is based on interpolation and the second is based on the idea of *sumsets*. First, we need the following well known lemma for special matrices known as Vandermonde matrices.

Lemma 3. Consider an $n \times n$ matrix M of the form :

$$M = \begin{bmatrix} 1 & \alpha_0 & \alpha_0^2 & \cdots & \alpha_0^{n-1} \\ 1 & \alpha_1 & \alpha_1^2 & \cdots & \alpha_1^{n-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \alpha_{n-1} & \alpha_{n-1}^2 & \cdots & \alpha_{n-1}^{n-1} \end{bmatrix}$$

Then,

$$\det(M) = \prod_{0 \le i < j \le n-1} (\alpha_j - \alpha_i)$$

Theorem 8. Let $\mathbb{F}$ be a field of zero or *large* characteristic. There exists distinct $\lambda_i \in \mathbb{F}$ such that for any $f(x) \in \mathbb{F}[x]$ and $i \in [r]$, we have :

$$f^i(x) = \sum_{j=0}^{r} c_{ij} \left(f(x) + \lambda_j\right)^r \qquad\qquad c_{ij} \in \mathbb{F}$$

Proof. Consider the polynomial $(f(x)+t)^r$. Here t is another different indeterminate. Using the binomial theorem, we can write :

$$(f(x)+t)^r = \sum_{i=1}^{r} \binom{r}{i} t^i f^{r-i} \tag{3.1}$$

As usually done in interpolation, we choose $r+1$ different λ_j and get $r+1$ equations from (1), by plugging in $t = \lambda_j$. This can be represented in the matrix form as $Ay = b$, where :

$$A = \begin{bmatrix} \binom{r}{0}\lambda_0^0 & \binom{r}{1}\lambda_0 & \cdots & \binom{r}{r}\lambda_0^r \\ \binom{r}{0}\lambda_1^0 & \binom{r}{1}\lambda_1 & \cdots & \binom{r}{r}\lambda_1^r \\ \vdots & \vdots & \vdots & \vdots \\ \binom{r}{0}\lambda_r^0 & \binom{r}{1}\lambda_r & \cdots & \binom{r}{r}\lambda_r^r \end{bmatrix}$$

$$y = \begin{bmatrix} 1 \\ f \\ \vdots \\ f^r \end{bmatrix} \qquad b = \begin{bmatrix} (f(x)+\lambda_0)^r \\ (f(x)+\lambda_1)^r \\ \vdots \\ (f(x)+\lambda_r)^r \end{bmatrix}$$

The determinant is invariant under the operation of adding scalar multiples of rows/columns to rows/columns respectively. This, coupled with the property of Vandermonde matrices, we get :

$$\det(A) = \prod_{i=0}^{r} \binom{r}{i} \prod_{0 \le i < j \le r} (\lambda_j - \lambda_i)$$

Since the λ_j's are distinct, $\det(A) \neq 0$. Thus, A is invertible. Hence, we have $y = A^{-1}b$. Let the $(i+1)$-th row in A^{-1} be $[c_{i0} \;\; c_{i1} \;\; \cdots \;\; c_{ir}]$. Then, we have :

$$f^i(x) = \sum_{j=1}^{r} c_{ij} \left(f(x) + \lambda_j\right)^r$$

□

The second representation uses the idea of ***sumsets***. The notion is central to the field of additive combinatorics [[Lovett, 2017] is a good survey for appplications of additive combinatorics to Theoretical Computer Science].

Definition 9. Let A and B be subsets of G, an additive group. The *sumset* of A and B is defined as :

$$A + B := \{g \in G : \exists\ a \in A \text{ and } b \in B, \text{such that } g = a + b\}$$

Usually, the sets that are considered are $\mathbb{Z}$, $\mathbb{Z}/n\mathbb{Z}$ or $\mathbb{R}$. When considering a field $\mathbb{F}$, we simply consider the underlying abelian group structure. Also, by nA, we denote the n-fold iterated sumset of A :

$$nA := \underbrace{A + A + \cdots + A}_{n \text{ times}}$$

We wish to construct a *small-support union* representation of a d-degree polynomial f as a sum of r-th powers, where r is a constant. The idea is to consider a set B such that rB covers $[d]$. In particular, $\exists$ a *unique* non-negative integer t such that :

$$(t-1)^r \leq \quad d+1 \quad \leq t^r \tag{3.2}$$

We consider the following set B :

$$B = \{a_i t^k \;\; | \; a_i \;\; \in [t-1], \;\; b \;\; \in [r-1]\}$$

For each t and k, we have a distinct element of B, hence $|B| = rt$. From (2), we get $t = O\left(d^{1/r}\right)$. So, $|B| = O\left(r \cdot d^{1/r}\right)$. Pick any integer $a \in [d]$. The *base-t* representation of a has at most r elements from B. Thus, $rB \supseteq [d]$.

Also, notice that the largest element of B is $(t-1) \cdot t^{r-1}$. Since, $(d+1) \leq t^r$, we have that, for any $\delta > 0$:

$$t \leq (1+\delta)\,(d+1)^{1/r}$$

Thus, for any $c \geq 0$, we have :

$$(t-1)\,t^{r-1} \leq c\,(d+1)$$

Thus, the largest element of rB is $r \cdot (t-1)\,t^{r-1} = O(d)$. We now prove a lemma that demonstrates how to use this setup to find a small-support union of f.

Lemma 4. Let $\mathbb{F}$ be a field of zero or large characteristic. Let $m := (t-1)\,t^{r-1}$. For any $f \in \mathbb{F}[x]$ of degree d, $\exists\ l_i$ supported on B and $c_i \in \mathbb{F}$ such that :

$$f(x) = \sum_{i=0}^{mr} c_i l_i^r$$

Proof. Write $l_i(\mathbf{z}_i, x) = \sum_{j \in B} z_{ij} x^j$, for distinct intermediates $z_{ij}\ \forall\ i, j$. We have $\deg_x(l_i) = m$. There are $mr+1$ polynomials Q_j over $|B|$ many variables of degree r such that we have :

$$l_i(\mathbf{z}_i, x)^r = \sum_{j \in [mr]} Q_j(\mathbf{z}_i)\, x^j \qquad \forall i \in [d]$$

Clearly, Q_j's are $\mathbb{F}$-linearly independent. Now, suppose $f = \sum_{i \in [d]} f_i x^i$. Define $\hat{f}$ to be a $mr \times 1$ row-vector over $\mathbb{F}$ and consider $A \in \mathbb{F}[\mathbf{z}]^{(mr+1)\times(mr+1)}$ as follows :

$$\hat{f} = [f_0 \ \ f_1 \ \ \cdots \ \ f_d \ \ 0 \ \ \cdots \ \ 0]$$

$$A = \begin{bmatrix} Q_0(\mathbf{z}_0) & Q_1(\mathbf{z}_0) & \cdots & Q_{mr}(\mathbf{z}_0) \\ Q_0(\mathbf{z}_1) & Q_1(\mathbf{z}_1) & \cdots & Q_{mr}(\mathbf{z}_1) \\ \vdots & \vdots & \vdots & \vdots \\ Q_0(\mathbf{z}_{mr}) & Q_1(\mathbf{z}_{mr}) & \cdots & Q_{mr}(\mathbf{z}_{mr}) \end{bmatrix}$$

We want to find the $z_{ij} = \alpha_{ij} \in \mathbb{F}$, and $\hat{c} = [c_0 \ \ c_1 \ \ \cdots \ \ c_{mr}] \in \mathbb{F}^{1\times mr}$ such that :

$$\sum_{i \in [mr]} c_i l_i(\overline{\alpha}, x)^r = \sum_{i \in [d]} f_i x^i$$

$$\iff \hat{c} \cdot A|_{\mathbf{z}=\overline{\alpha}} \cdot \begin{bmatrix} 1 \\ x \\ \vdots \\ x^{mr} \end{bmatrix} = \hat{f} \cdot \begin{bmatrix} 1 \\ x \\ \vdots \\ x^{mr} \end{bmatrix}$$

$$\iff \hat{c} \cdot A|_{\mathbf{z}=\overline{\alpha}} = \hat{f}$$

Since, $\mathbf{z}_i$ are distinct variables, the first column of A consists of different variables at each co-ordinate and the Q_j's are $\mathbb{F}$-linearly independent, therefore $\det(A)$ is not the identically zero polynomial. Thus, by the Schwartz-Zippel lemma, for random α_{ij}, plugging in $z_{ij} = \alpha_{ij}$ gives us an invertible matrix $A|_{\mathbf{z}=\overline{\alpha}}$. Since a random $\overline{\alpha}$ works, at least one such $\overline{\alpha}$ exists. Pick such $\overline{\alpha}$.Thus, we get :

$$\hat{c} = (A|_{\mathbf{z}=\overline{\alpha}})^{-1} \cdot \hat{f}$$

Thus, we get our required identity :

$$f(x) = \sum_{i \in [mr]} c_i \cdot l_i(\overline{\alpha}, x)^r$$

□

3.2 Analysis of the *Support-Union* measure for specific cases

We recall the *Sum of Powers* model representation over a univariate polynomial ring $R[x]$:

$$f = \sum_{i=1}^{s} c_i l_i^r \tag{3.3}$$

We recall the *support-union* measure, denoted by $U_R(f,r,s)$, :

$$U_R(f,r,s) := \begin{cases} \min_{f=\sum_{i=1}^{s} c_i l_i^r} \left| \bigcup_{i=1}^{s} \text{supp}(l_i) \right| & \text{if such a representation exists for fixed } r \text{ and } s \\ \infty & \text{otherwise} \end{cases}$$

We wish to investigate this measure for the polynomial family : $f_d := (x+1)^d$. First we make a couple of small observations about this measure :

- In the minimal representation, consider the number of distinct monomials used in the representation. This is exactly the measure defined above. Note that, we have the following very simple inequality for *sumset* estimates :

  $$|A+B| \leq |A| \cdot |B|$$

  By, induction :

  $$|nA| \leq |A|^n$$

  Let S be the set of indices of distinct monomials in a univariate g. To consider the number of monomials in g^k, we need to consider the set kS. But, this is upper bounded by $|S|^k$. By a simple extension, this argument shows :

  $$\begin{aligned} & U_R(f,r,s)^r \geq \left|\text{supp}(f)\right| \\ \iff & U_R(f,r,s) \geq \left|\text{supp}(f)\right|^{1/r} \end{aligned}$$

  This the first trivial lower bound that we obtain on the measure U for any f.

- For large s, i.e for any $s \geq c(d+1)$, for any $c > r$, Lemma 4 shows that, $U_{\mathbb{F}}(f_d,r,s) \leq O\left(d^{1/r}\right)$. The above argument gives a matching lower bound. So, for large s, we have $U_{\mathbb{F}}(f_d,r,s) = \Theta\left(d^{1/r}\right)$.

Notice that one might wonder if the case that $U_R(f,r,s) = \infty$ is ever possible ? Of course, we have shown that for $s \geq r+1$, this measure is finite by Theorem 8. However, it is indeed possible that such a representation might not exist. To see this, consider the following, which is a polynomial analogue of the legendary *Fermat's Last Theorem*.

Theorem 9. [Polynomial Fermat's Last Theorem] Let $f(x), g(x), h(x)) \in \mathbb{C}[x]$ be co-prime polynomials satisfying

$$f^n + g^n = h^n$$

for some $n \geq 3$. Then, these polynomials are constant.

Proof. To prove this, we need to prove the following Lemma, known as the Mason-Stother's Theorem.

Lemma 5. [Mason Stother's Theorem] Suppose that $f(x)$, $g(x)$ and $h(x)$ are co-prime polynomials and not all of them are constant. If $f+g+h=0$, then :

$$\max(\deg(f),\deg(g),\deg(h)) \leq Z(fgh)-1$$

where $Z(f)$ denotes the number of distinct complex roots of f.

Proof. Note that if $f(x)=(x-\alpha_1)^{e_1}(x-\alpha_2)^{e_2}\cdots(x-\alpha_r)^{e_r}$ then :

$$\gcd\left(f,f'\right)=(x-\alpha_1)^{e_1-1}(x-\alpha_2)^{e_2-1}\cdots(x-\alpha_r)^{e_r-1}$$

Thus,

$$\deg\left(\gcd\left(f,f'\right)\right)=\deg(f)-Z(f)$$

Notice that taking the derivative on both sides of $f+g+h=0$, and doing some trivial compuation, one can obtain $fh'-f'h=f'g-fg'$. Without loss of generality, assume f and g are non-constant. Also, since f and g are co-prime, we must have $fg'-f'g\neq 0$.

Note that $\gcd(f,f')$ and $\gcd(g,g')$ divide $f'g-fg'$ and from that and the discussion in the earlier paragraph, we can conclude $\gcd(h,h')$ divides $f'g-fg'$. Also, we have :

$$f,g,h \text{ co-prime} \implies \gcd\left(f,f'\right),\gcd\left(g,g'\right),\gcd\left(h,h'\right) \text{ are co-prime.}$$

So $\gcd(f,f')\times\gcd(g,g')\times\gcd(h,h')$ divides $f'g-g'f$. Thus, we can conclude :

$$(\deg(f)-Z(f))+(\deg(g)-Z(g))+(\deg(h)-Z(h)) \leq \deg\left(f'g-fg'\right) \leq \deg(f)+\deg(g)-1$$
$$\implies \deg(h) \leq Z(f)+Z(g)+Z(h)-1 \leq Z(fgh)-1$$

Similarly, we can obtain exactly the same inequalities for f and g. This gives us :

$$\max(\deg(f),\deg(g),\deg(h)) \leq Z(fgh)-1$$

□

We will now prove Theorem 9 using the above lemma. Assume that such a representation is possible. Then, we have the following equations, using Lemma 5 for f^n, g^n and $-h^n$:

$$n\deg(f) \leq Z\left(f^n g^n h^n\right)-1 \leq Z(fgh)-1 \leq \deg(f)+\deg(g)+\deg(h)-1$$

$$n\deg(g) \leq Z\left(f^n g^n h^n\right)-1 \leq Z(fgh)-1 \leq \deg(f)+\deg(g)+\deg(h)-1$$

$$n\deg(h) \leq Z\left(f^n g^n h^n\right)-1 \leq Z(fgh)-1 \leq \deg(f)+\deg(g)+\deg(h)-1$$

Adding all the above inequalities, we get :

$$n\left(\deg(f)+\deg(g)+\deg(h)\right) \leq 3\left(\deg(f)+\deg(g)+\deg(h)\right)-3$$

which is a contradiction, since $n \geq 3$! Thus, our proof is complete. □

Notice that for the representation :

$$(x+1)^{3r} = l_1^{6r} + l_2^{6r} \qquad r \geq 1$$

Theorem 9 is applicable. This is because if $l_1 = (x+1)^{e_1} g(x)$ and $l_2 = (x+1)^{e_2} h(x)$ ($\gcd(g(x), x+1) = \gcd(h(x), x+1) = 1$), then $e_1 = e_2$, otherwise we'll get $(x+1)^{e_3}$ divides $h(x)$, where $e_3 \geq 1$. which is clearly a contradiction ! Thus, without loss of generality, we may assume that l_1, l_2 and $x+1$ are co-prime . Thus, Theorem 9 is applicable and there is no such representation. Hence,

$$U_{\mathbb{C}}\left(f_{3r}, 6r, 2\right) = \infty$$

for any $r \geq 1$.

3.2.1 Upper and Lower Bounds

For $s = 2$, we present a strong lower bound of $\Omega\left(d/r\right)$ and for $s = (r+1)$ (which is a case of small s), we present an upper bound of $O\left(d/r + r\right)$ on $U_{\mathbb{F}}\left(f_d, r, s\right)$ [Dutta, Saxena, and Thierauf, 2020].

Theorem 10. [Upper Bound] For any $d \in \mathbb{N}$ and $r \leq d$, we have $U_{\mathbb{F}}\left((x+1)^d, r, r+1\right) \leq \frac{d}{r} + r$.

Proof. Suppose $d = r \cdot k + t$, where $t \equiv d \pmod r$ and $k \in \left[\frac{d}{r}\right]$. From Theorem 8, it follows that $\exists\, c_i, \lambda_i \in \mathbb{F}$, such that :

$$\begin{aligned}
(x+1)^d &= \left((x+1)^k\right)^r \cdot (x+1)^t \\
&= \left((x+1)^k\right)^r \cdot \left(\sum_{i\in[r]} c_i \left((x+1)^t + \lambda_i\right)^r\right) \\
&= \sum_{i\in[r]} c_i \left((x+1)^{t+k} + \lambda_i (x+1)^k\right)^r \\
&= \sum_{i\in[r]} c_i l_i^r
\end{aligned}$$

Here, $l_i = (x+1)^{t+k} + \lambda_i (x+1)^k$. We have :

$$\left|\bigcup_{i\in[r]} \operatorname{supp}\left(l_i\right)\right| \leq t+k+1 \leq \frac{d}{r} + r$$

Hence, we are done. □

As mentioned before, $U_{\overline{\mathbb{F}}}(f,r,s) \leq U_{\mathbb{F}}(f,r,s)$, so for the sake of proving a lower bound on the measure, we can assume, without loss of generality, that $\mathbb{F}$ is algebraically closed. Let us assume we are working over $\mathbb{C}$. Note that, over $\mathbb{C}$, the r-th root of -1 exists, for any r. We present the following lower bound :

Theorem 11. [Lower Bound] For any $d \geq 1$ and any $r \geq 2$, we have :

$$U_{\mathbb{C}}(f_d,r,2) \geq \begin{cases} \lceil d/r \rceil + 1 & \text{if } r|d \text{ or } r = 2 \\ \infty & \text{otherwise} \end{cases}$$

To prove this theorem we prove a couple of lemmas first.

Lemma 6. For a fixed $d \geq 1$ and $r \geq 3$, if $(x+1)^d = l_1^r - l_2^r$ for some $l_i \in \mathbb{C}[x]$, then l_1 and l_2 share a non-tivial g.c.d .

Proof. Assume l_1 and l_2 are co-prime. We have the following factorization :

$$l_1^r - l_2^r = \prod_{i=0}^{r-1} \left(l_1 - \zeta_r^i l_2\right)$$

where ζ_r is a primitive r-th root of unity. Since, $l_1^r - l_2^r = (x+1)^d$, $(x+1)$ must divide some of the factors. Suppose, it divides, for $i \neq j$, $\left(l_1 - \zeta_r^i l_2\right)$ and $\left(l_1 - \zeta_r^j l_2\right)$. Thus, $(x+1)$ divides their linear combinations, particularly $(x+1)\,|l_1, l_2$. Since, l_1 and l_2 are co-prime, it implies that $\left(l_1 - \zeta_r^i l_2\right) = c \cdot (x+1)^d$ for a unique $i \in [r-1]$ and for every $j \neq i$, $\left(l_1 - \zeta_r^j l_2\right)$ must be a constant. Again, by taking an appropriate linear combination, it implies that l_1 and l_2 are constants. This is a contraction, since we assumed that $d \geq 1$. □

An immediate corollary is as follows which resolves Theorem 11, for $r \geq 3$.

Lemma 7. Consider $3 \leq r \leq d$. Then, $(x+1)^d = l_1^r - l_2^r$ *iff* $r|d$. Also, $\exists\, c_1$ and $c_2 \in \mathbb{F}$, such that $l_i = c_i (x+1)^{d/r}$.

Proof. Denote by $g(x)$, the non-trivial g.c.d of l_1 and l_2 (Lemma 6 guarantees it's existence). This implies $g^r | (x+1)^d$. Thus, $g(x)$ is a power of $(x+1)$. By dividing out the g.c.d, we get a new equation :

$$(x+1)^{d'} = \hat{l}_1^r - \hat{l}_2^r$$

which has exactly the similar form as the initial equation. Thus, by doing this repeatedly, we get that $r|d$. Also, this implies that $\exists\, c_1$ and c_2 such that $l_i = c_i (x+1)^{d/r}$. □

Now, we complete the proof of Theorem 11.

Proof. The case for $r \geq 2$ is resolved by Lemma 7. For $r = 2$, we have :

$$(x+1)^d = (l_1 + l_2) \cdot (l_1 - l_2)$$

Thus, $\exists\, c_1, c_2 \in \mathbb{F}$ such that $(l_1 + l_2) = c_1 (x+1)^k$ and $(l_1 - l_2) = c_2 (x+1)^{d-k}$. Clealy,

$$U_{\mathbb{C}}(f_d, 2, 2) \geq \max(k+1, d-k+1) \geq \lceil d/2 \rceil + 1$$

This concludes the proof. □

3.3 The *Sparsity-Sum* Measure is large for *random* polynomials

Recall the *Sparsity-Sum* measure for the SOP representation of a polynomial f over a ring R, denoted by $S_R(f, r, s)$:

$$S_R(f,r,s) := \begin{cases} \min\limits_{f=\sum_{i=1}^{s} c_i l_i^r} \left(\sum\limits_{i=1}^{s} |l_i|_1 \right) & \text{if such a representation exists for fixed } r \text{ and } s \\ \infty & \text{otherwise} \end{cases}$$

From Theorem 8, we get that both the measures S and U are finite if $s \geq r+1$. An important property of the measure S is that S is large for *random* polynomials. In particular, for a random polynomial, we can consider it's coefficients being represented by random variables that are algebraically independent. Now, consider the SOP representation f for random f:

$$f = \sum_{i=1}^{s} c_i l_i^r \tag{3.4}$$

This shows that we can consider each l_i having coefficients represented by polynomials in the random variables making up the coefficient of f. Before we continue, we want to define what we mean by the *transcendence* degree of a set of polynomials.

Definition 10. The *transcendence degree* of a set of polynomials $S = \{f_1, f_2, ..., f_m\} \in \mathbb{F}[x_1, x_2, ..., x_n]$ is defined as the maximal number of algebraically independent polynomials in S.

We outline the formal proof below :

Lemma 8. Consider a *random* polynomial f and it's SOP representation as in (4). Then,

$$\sum_{i=1}^{s} |l_i|_1 \geq \Omega(|f|_1)$$

implying that $S_{\mathbb{F}}$ is large.

Proof. As mentioned in the prior discussion, we can view the l_i as univariate polynomials in x, with coefficents in the ring $\mathbb{F}\left[y_{i1}, y_{i2}, .., y_{it_i}\right]$, where the transcendence degree of the coefficent polynomials in l_i is t_i (this is achieved through the variable reduction lemma of [Pandey, Saxena, and Sinhababu, 2018, Lemma 2.8]). This clearly implies $|l_i|_1 \geq t_i$. Coefficients of l_i^r are generated by algebraic combinations of coefficients of l_i, thus the transcendence degree of the coefficients of l_i^r is at most $t_i + 1$. Since, the coefficients of f are represented by random variables that are algebraically independent, the transcendence degree of the coefficients of f is $|f|_1$. Thus, we get :

$$\sum_{i=1}^{s} (|l_i|_1 + 1) \geq \sum_{i=1}^{s} (|t_i|_1 + 1) \geq |f|_1$$

by which we obtain $S_{\mathbb{F}}(f, r, s) \geq \Omega(|f|_1)$. □

Open Question 1 : Can we show $U_{\mathbb{F}}(f, r, s) \geq \Omega(|f|_1)$ for random f ? The proof from Lemma 8 does not directly go through .

3.4 The Main Conjecture and an Unconditional Lower Bound

The goal of this thesis is to study the measures $U_{\mathbb{F}}(\cdot)$ and $S_{\mathbb{F}}(\cdot)$ for very specific special polynomials $f_d = (x+1)^d$ and $g_d := \sum_{i=0}^{d} 2^{i^2} x^i$. The aim is to see the behavior of the measure with changing d. We had outlined the resolution of certain special cases in the earlier section. We now make the central conjecture that is the focus of this thesis.

Conjecture 2. Fix any arbitrarily small growing prime function $r(\cdot)$ over $\mathbb{N}$ such that $r(d) \leq \log^* d$-th prime . Then, $\exists$ a constant $\delta > 0$ such that $U_{\mathbb{F}}(g_d, r(d), s(d)) \geq \Omega\left(d/r^{\delta}\right)$ for an arbitrarily small growing function $s(d) = d^{o(1)}$.

Restricting $r(\cdot)$ to a prime function is not a very stringent requirement because from Bertrand's postulate, $\exists$ a prime in the interval $[n, 2n]$ for every $n \geq 1$. So, for any asymptotic argument, this extra requirement does not lead to any loss of generality, because of the *sandwich lemma*.

We expect this conjecture to be true even for the family f_d or the family $h_d = \prod_{i=1}^{d} (x-i)^d$ [Dutta, Saxena, and Thierauf, 2020]. It is noteworthy that the conjectured lower bound is very tight for the family $(x+1)^d$ because we had outlined in Theorem 8 that $U_{\mathbb{F}}(f_d, r, r+1) \leq O(d/r)$ for d and small r. The main aim of the next chapter will be to shed light on the *explicitness* of the polynomial families f_d and g_d and to show that Conjecture 2, if true, implies that Valiant's hypothesis is true.

We now focus our attention to localized integer rings. First, let us define each of those two terms.

Definition 11. Consider a ring R. A subset $S \subset R$ is said to be *multiplicatively closed* if $1 \in S$ and $ab \in S \ \forall \ a, b \in S$. Let $S \subset R$ be a multiplicatively closed set. Then, the relation :

$$(a,s) \sim \left(a',s'\right) \qquad :\iff \qquad \text{there is an element } u \in S \text{ such that } u\left(as' - a's\right) = 0.$$

is an equivalence relation on $R \times S$. We denote the equivalence class of a pair $(a,s) \in R \times S$ as $\frac{a}{s}$. The set of all equivalence classes

$$S^{-1}R := \left\{ \frac{a}{s} : a \in R, s \in S \right\}$$

is called *localization* of R at S.

As a primary example of localization, and the most relevant in our case, consider when $P \subset R$ is a prime ideal. The set $S = R \setminus P$ is multiplicatively closed, because $a \notin P$ and $b \notin P$, implies $ab \notin P$ as P is a prime ideal. the resulting localization $S^{-1}R$, denoted as R_P, is called the localization of R at the prime ideal P.

Definition 12. The *ring of integers* of an algebraic number field K (finite extension of $\mathbb{Q}$) is the set of all integral elements contained in K. An *integral element* is the root of a monic polynomial with integer coefficients. This ring is usually denoted by $\mathcal{O}_K$.

The simplest possible template to keep in mind is the ring $\mathbb{Z}$. Also, since any integer $\in K$ and is the root a monic (linear) polynomial, the ring $\mathbb{Z}$ is always a subring of $\mathcal{O}_K$.

Let $\mathbb{P}$ be a prime ideal of $\mathcal{O}_K$. Consider the localization $(\mathcal{O}_K)_{\mathbb{P}}$. Notice that this is strictly larger than $\mathcal{O}_K$. For example, if we consider the localized ring $\mathbb{Z}_p$ (**not** the ring of p-adic integers), it a subring of $\mathbb{Q}$ that has all fractions except those whose denominators are divisible by p.

Now, we state the unconditional lower bound, proven in [Dutta, Saxena, and Thierauf, 2020] :

Theorem 12. Consider the family $g_d := \sum_{i=0}^{d} 2^{i^2} x^i$. Fix any odd prime r and any $s \geq 1$. Fix a number field K and prime ideal $\mathbb{P}$ over $\mathcal{O}_K$ such that $\mathbb{P} \Big| \langle r \rangle_{\mathcal{O}_K}$. Then,

$$U_{(\mathcal{O}_K)_{\mathbb{P}}}(g_d, r, s) \geq \Omega(d)$$

Proof. It is clear that $2^{i^2} \notin \langle r \rangle_{(\mathcal{O}_K)_{\mathbb{P}}}$ for any $i \in [d]$. To see this, assume, for the sake of contradiction, that $2^{i^2} \in \langle r \rangle_{(\mathcal{O}_K)_{\mathbb{P}}}$ for some $i \in [d]$. Then, this implies that $\exists\ s \in (\mathcal{O}_K)_{\mathbb{P}}$

such that $2^{i^2} = sr$. However, this implies that $s = 2^{i^2}/r$ which cannot be in $(\mathcal{O}_K)_{\mathbb{P}}$ as $\mathbb{P}\Big|\langle r\rangle_{\mathcal{O}_K}$. Thus, we get :

$$\begin{aligned}
& g_d(x) = \sum_{i=1}^{s} l_i^r \\
\implies & g_d \pmod{\langle r\rangle_{(\mathcal{O}_K)_{\mathbb{P}}}} = \sum_{i=1}^{s} l_i(x^r) \pmod{\langle r\rangle_{(\mathcal{O}_K)_{\mathbb{P}}}} \\
\implies & \left|\bigcup_{i=1}^{s} \operatorname{supp}(l_i(x^r))\right| = d+1 \\
\implies & \left|\bigcup_{i=1}^{s} \operatorname{supp}(l_i)\right| = d+1
\end{aligned}$$

Hence, we are done. □

We can also prove that the same conclusion holds for $f_d = (x+1)^d$, for specific forms of d. Fix $d = (r-1)r^{l-1} + (r-1)r^{l-2} + \cdots + (r-1)$, for some prime r. The following result is key to the proof.

Lemma 9 (Lucas' Theorem)**.** For non-negative m and n and a prime p, the following relation holds :

$$\binom{m}{n} \equiv \prod_{i=1}^{k} \binom{m_i}{n_i} \pmod{p}$$

where $m_i = \sum_{i=0}^{k} m_i p^i$ and $n_i = \sum_{i=0}^{k} n_i p^i$ are the base p representations of m and n respectively.

For the sake of completion, we include a proof using generating functions, due to [Fine, 1947].

Proof. From the property of binomial coefficients of the form $\binom{p}{i}$ for a prime p, we get :

$$(1+X)^p \equiv 1 + X^p \bmod p$$

By induction, for every non-negative integer i,

$$(1+X)^{p^i} \equiv 1 + X^{p^i} \bmod p$$

Now, notice that :

$$\begin{aligned}
\sum_{m=0}^{n} \binom{m}{n} X^n = (1+X)^m &= \prod_{i=0}^{k} \left((1+X)^{p^i}\right)^{m_i} \\
&\equiv \prod_{i=0}^{k} \left(1+X^{p^i}\right)^{m_i} \mod p \\
&\equiv \prod_{i=0}^{k} \left(\sum_{n_i=0}^{m_i} \binom{m_i}{n_i} X^{n_i p^i} \right) \mod p \\
&\equiv \prod_{i=0}^{k} \left(\sum_{n_i=0}^{p-1} \binom{m_i}{n_i} X^{n_i p^i} \right) \mod p \\
&\equiv \sum_{n=0}^{m} \left(\prod_{i=0}^{k} \binom{m_i}{n_i} \right) X^n \mod p
\end{aligned}$$

Comparing and equating coefficients, the proof is complete.

□

Notice that, since $\binom{m}{n} = 0$, whenever $m < n$, looking at the base r representation of the given form of d and applying Lemma 9, immediately gives us :

Theorem 13. For a prime r, suppose $d = \sum_{i=0}^{l} a_i r^i$ where $0 \le a_i \le r-1$. Then, we have $|(x+1)^d \bmod r|_1 = \prod_{i=0}^{l} (a_i + 1)$.

Using this theorem, the desired conclusion is obvious by using the exact steps used in the proof of Theorem 12.

Chapter 4

Depth Reduction, Explicitness Criterion and Valiant's Hypothesis

In this chapter, we focus on the striking connection between Conjecture 2 and Valiant's Hypothesis (Theorem 19). To do so, we use depth-reduction ideas due to [Valiant et al., 1983] and [Agrawal and Vinay, 2008]. We give an outline of reduction of a circuit to an equivalent *log-depth* (logarithmic in the degree of polynomial computed) circuit by [Valiant et al., 1983] which only causes a polynomial blow up in size. We use this fact to point out that in order to prove a lower bound, it is enough to assume that our circuit is already in the form given in [Valiant et al., 1983], which has certain special properties. We mention these properties and call such circuits as *universal circuits* or *circuits in normal-form*.

Once, we have circuits in normal form, we use the proof technique in [Agrawal and Vinay, 2008], particularly by slicing the circuit at a certain depth t and then analyzing each of the top and bottom parts separately and optimizing over t.

In this chapter, we also explain in detail what we mean by *explicitness* of a polynomial family. We do so by considering the *Counting Hierarchy* introduced by [Wagner, 1986] and using the setup in [Bürgisser, 2009] and [Koiran, 2011]. Roughly, we want the j-th bit of the i-th coefficient in each polynomial of the family $\{f_n\}_{n\in\mathbb{N}}$ be computable in reasonable time (i.e they belong to a reasonably small complexity class), as a function of the parameter n, i, j.

4.1 Depth Reduction - Outline of Reduction to log depth

Depth-reduction results allow us to show that we can simulate general circuits with *low-depth* circuits. The first significant result [Valiant et al., 1983], [Allender et al., 1998] in this area is as follows :

Theorem 14. Let f be an n-variate degree d polynomial computed by an arithmetic circuit C of size s. Then, there is an arithmetic circuit C' computing f and has size $s' = \text{poly}(s, n, d)$ and depth $O(\log d)$.

We give a short proof sketch below, following [Saptharishi, 2020].

Proof Sketch. As pointed out in [Saptharishi, 2020], we can assume, without loss of generality, that C is a homogeneous circuit, all multiplication gates have fan-in at most 2 and that the degree of the right child of any multiplication gate is at least as

large as the degree of its left child (such circuits are called *right-heavy* circuits). For any gate u in $\mathcal{C}$, let us denote by $[u]$, polynomial computed at the gate u. We also denote by u_L and u_R, the left and right child of u, respectively. The following gate quotients are defined :

Definition 13. For any pair of gates u, v, we define the polynomial $[u : v]$ as follows :

- If u and v are the same nodes, then $[u : v] = 1$.
- If u is a leaf and $u \neq v$, then $[u : v] = 0$.
- If $u = u_1 + u_2$ (i.e u is an addition gate), then $[u : v] = [u_1 : v] + [u_2 : v]$.
- If $u = u_1 \times u_2$ (i.e u is a multiplication gate), then $[u : v] = [u_1] \cdot [u_2 : v]$.

One can easily show since we are working with a homogeneous circuit $\mathcal{C}$, $[u : v]$ is a homogeneous polynomial of degree $\deg(u) - \deg(v)$ respectively. We now define a notion that is central to our proof.

Definition 14. For any parameter m, define the *frontier at degree m*, as :

$$\mathcal{F}_m = \{v : \deg(v) \geq m, \ \ \deg(v_L), \deg(v_R) < m\}$$

So, $\mathcal{F}_m$ are the deepest nodes in the circuit that have degree at least m.

Since, we are working with homogeneous circuits, note that the degree of the polynomial computed by a parent can be greater than that of the maximum degree of polynomials computed by the children only if the gate is a multiplication gate. So, all frontier gates are multiplication gates. The following is the crucial lemma that is used to prove depth reduction, which can be proved by induction on the depth of the node u.

Lemma 10. Suppose $\mathcal{C}$ is a homogeneous, right heavy circuit. Let m be a parameter such that $\deg(u) \geq m$. Then,

$$[u] = \sum_{w \in \mathcal{F}_m} [u : w][w] \tag{4.1}$$

Also, if u, v are nodes such that $\deg(u) \geq m > \deg(v)$, then

$$[u : v] = \sum_{w \in \mathcal{F}_m} [u : w][w : v] \tag{4.2}$$

The idea, thereafter, is to compute $[u]$ and $[u : v]$ from nodes of lower degree. adopting a top-down approach. This approach is similar to that of [Allender et al., 1998] but contrary to that of [Valiant et al., 1983], who followed a bottom-up approach instead. We consider any u and fix $m = \frac{\deg(u)}{2}$. Denote $\mathcal{F}_m$ as $\mathcal{F}(u)$. Using

Lemma 10, we can write $[u]$ as :

$$[u] = \sum_{w \in \mathcal{F}(u)} [u : w] \cdot [w_L] \cdot [w_R]$$

All the terms on the R.H.S has degree at most $\frac{\deg(u)}{2}$. Note that $[u]$ is an addition gate and the multiplication gates feeding into it have fan-in 3 and the gate itself has fan-in at most s. Now that we have computed $[u]$, we need to figure out how to compute $[u : v]$. Since, $[u : v]$ is a homogeneous polynomial of degree $\deg(u) - \deg(v)$, we set $m = \frac{\deg(u)-\deg(v)}{2}$ and consider $\mathcal{F}_m$. To avoid any blowup in degree of the left child, Lemma 10 is applied twice to obtain :

$$[u : v] = \sum_{w \in \mathcal{F}_m} \sum_{x \in \mathcal{F}(w_L)} [u : w] \cdot [w_L : x] \cdot (x_L) \cdot (x_R) \cdot [w_R : v]$$

The terms of R.H.S has degree at most m. Also, note that the multiplication gates feeding into $[u : v]$ have fan-in 5 and the gate itself has fan-in at most s^2.
As the degree halves at every level, it is obvious that the resulting circuit $\mathcal{C}'$ that is built in the top-down fashion has depth $O(\log d)$. For further proof details, see [Saptharishi, 2020]. □

4.2 The Universal/Normal Form Circuit

The log-depth reduction procedure allows us to convert a circuit $\mathcal{C}$ to a *shallow* circuit $\mathcal{C}'$ having the following important properties :

1. alternative layers of multiplication and addition gates with the root (top-gate) being an addition gate
2. below each multiplication layer, the associated polynomial degree at least halves
3. fan-in of each multiplication gate is at most 5
4. depth of the circuit $\mathcal{C}'$ is at most $O(\log d)$, where d is the degree of the polynomial computed by the circuit $\mathcal{C}$.

Definition 15. A circuit $\mathcal{C}$ that has the 4 properties listed above is called a **universal cicuit/normal-form circuit**.

We call this circuit *universal* because any algebraic circuit is expressible in this form with only a polynomial blow-up in size. Each homogeneous part of a circuit of size s computing a polynomial of degree d can be computed by another circuit of size at most $s' = O\left(sd^2\right)$ [Strassen, 1973]. The depth-reduction result shows that we can compute an equivalent log-depth circuit of size at most $\left(s'^3\right)$. Hence, the resulting *universal* circuit has size at most $O\left(s^3 d^6\right)$ which is a poly blow-up.

This notion will be important later in this chapter when we draw the connection between Conjecture 2 and Valiant's Hypothesis, since we will mostly assume that we already have a circuit in normal-form.

4.3 Counting Hierarchy and Explicitness of Polynomial families

In Chapter 2, when we had outlined the usual template that is followed by lower bound proofs, we had mentioned the importance of finding an *explicit* polynomial family f such that the measure is large on that family. Here, we shall shed more light on what we mean by the *explicitness* criterion.

4.3.1 The Counting Hierarchy

The counting hierarchy was introduced by [Wagner, 1986] in order to classify the complexity of specific combinatorial problems where counting is involved. Let us make a few notations clear first.

Consider a pairing function denoted by $\langle , \rangle$:

$$\begin{aligned} \{0,1\}^* \times \{0,1\}^* &\longrightarrow \{0,1\}^* \\ (x,y) &\longrightarrow \langle x,y \rangle \end{aligned}$$

One simple way to construct a pairing function is to duplicate each bit and insert 01 in between. We now define a counting operator **C** to make notation less cumbersome.

Definition 16. Let K be a complexity class. We define $\mathbf{C} \cdot K$ to be the set of all languages A such that $\exists$ a language $B \in K$, a polynomial p and a polynomial-time computable function $f : \{0,1\}^* \to \mathbb{N}$ such that $\forall\ x \in \{0,1\}^*$:

$$x \in A \iff \left| \left\{ y \in \{0,1\}^{|p(x)|} \mid \langle x,y \rangle \in B \right\} \right| > f(x)$$

Definition 17. **[The Counting Hierarchy / CH]** The k-th level $C_k P$ of the counting hierarchy is recursively defined by $C_0 P := P$ and $C_{k+1} P := \mathbf{C} \cdot C_k P$. Then, we define :

$$\mathbf{CH} := \bigcup_{i=0}^{\infty} C_k P$$

This construction is very similar to that of the Polynomial Hierarchy (**PH**), whose constituting complexity classes are constructed by iterative and alternating application of the operators $\exists$ and $\forall$ on the class P. Since, it is obvious that $\exists \cdot K \subseteq \mathbf{C} \cdot K$ and $\forall \cdot K \subseteq \mathbf{C} \cdot K$, for any complexity class K, we get that $\mathbf{PH} \subseteq \mathbf{CH}$.

4.3.2 The meaning of explicitness

To define explicitness in our setting, let us begin by defining which integer sequences are *definable* in the Counting Hierarchy, following [Bürgisser, 2009] and [Koiran,

2011]. Let us consider sequences of integers $s(n,k)$, defined for n,k in unary $\in \mathbb{N}$ and a polynomially bounded function q, such that $0 \leq k \leq 2^{q(n)}$ and :

$$\forall n > 1,\ \forall k \leq 2^{q(n)} \quad |s(n,k)| \leq 2^{2^{n^c}}$$

for some constant c. When we think of n and k being represented in unary, sequences $s = (s(n,k))$ obeying the aforementioned inequality are referred to as sequences of *exponential bitsize*.

Let $|s| := (|s(n,k)|)$ denote the sequence of absolute values of s. Consider the following languages associated with a sequence $s = s(n,k)$ of exponential bitsize :

$$\text{Sgn}(a) \ := \ \{(1^n,k) \mid s(n,k) \geq 0\}$$

$$\text{Bit}\,(|a|) \ := \ \{(1^n,k,j,b) \mid \text{the } j\text{-th bit of } |s(n,k)| \text{ equals } b\}$$

Definition 18. **[CH** definability] A sequence s of integers of exponential bitsize are called *definable* in the Counting Hierarchy *iff* Sgn(s) and Bit $(|s|) \in$ **CH**. If both Sgn(s) and Bit $(|s|) \in$ **CH**/Poly, then we say that s is definable in **CH**/Poly.

Here, **CH**/Poly is simply the *nonuniform* version of the class **CH** (i.e with poly advice).

Definition 19. **[Explicitness]** We will call a polynomial family $\{f_n\}_{n\in\mathbb{N}}$ to be ***CH**-explicit* when the coefficient of the polynomial are definable in **CH**.

The intuition here is that the j-th bit of the k-th coefficent should be computable efficiently. If we write f_n as $f_n = \sum_k s(n,k)\bar{x}^k$, it is easy to see why the above definition of explicitness is extremely natural.

4.3.3 The Kronecker and Inverse Kronecker Maps

Let $\mathbb{F}_{\leq d}[x]$ (respectively $\mathbb{F}_{\leq d}[\bar{x}]$) be the ring of univariate (respectively, k-variate) polynomials of *individual* degree $\leq d$. The naive Kronecker map $\phi_{k,d}$ is an surjective homomorphism from $\mathbb{F}[\bar{x}]$ to $\mathbb{F}[x]$ which separates the k-variate monomials with maximum individual degree d. The map is defined as follows :

$$\phi_{k,d} \ : \ x_i \longrightarrow x^{(d+1)^{i-1}}$$

$\forall\ i \in [1,k]$. The most important property of $\phi_{k,d}$ encapsulated in the following lemma :

Lemma 11. $\phi_{k,d}$ givers distinct weights to distinct monomials of maximum individual degree d.

Proof. Consider a monomial of the form $x_1^{e_1}x_2^{e_2}\cdots x_k^{e_k}$. Each $e_i \leq d$. The map $\phi_{k,d}$ acts on the vector $(e_1, e_2, ..., e_k)$ as follows :

$$\phi \ : \ (e_1, e_2, ..., e_k) \longrightarrow \sum_{i=1}^{k} e_i \cdot (d+1)^{i-1}$$

The domain element is nothing but taking the base $(d+1)$-representation of an integer where the integer is precisely the image on the right. For a fixed base r, every integer has a unique base r-representation. This proves the claim. □

Note that the R.H.S is largest when the individual degree of every variable in the monomial is d. Thus, the above proof, when linearly extended to the case of polynomials shows that every polynomial with individual degree at most d gets mapped to a univariate polynomial of degree at most $d + d(d+1) + \cdots + d(d+1)^{k-1} = (d+1)^{k-1}$. Thus, $\phi_{k,d}$ is an bijective map between $\mathbb{F}_{\leq d}[\overline{x}]$ and $\mathbb{F}_{\leq (d+1)^k - 1}[x]$.

Now, we need to construct an inverse of this map. Intuitively, it is clear what the map should be. Yet, for the sake of formality, let us go through the steps. Suppose we want a map such that each x^i for $i \in [d]$ gets mapped to a distinct k-variate monomials of individual degree at most d_k. This inverse map, which we denote by $\psi_{k,d}$ is a map from $\mathbb{F}_{\leq d}[x]$ to $\mathbb{F}_{\leq d_k}[\overline{x}]$. We define $\psi_{k,d}$ as :

$$\psi_{k,d} : \ x^i \ \longrightarrow \overline{x}^{\text{base}_{d_k+1}(i)}$$

Here $\text{base}_{d_k+1}(i)$ is the k-tuple $(i_1, i_2, ..., i_k)$ such that $i = \sum_{j=1}^{k} i_j (d_k+1)^{j-1}$. As before, we can linearly extend the map to polynomials of degree $\leq d$. Therefore, it suffices to choose a d_k such that $(d_k+1)^k - 1 \geq d \geq d_k^k - 1$. We choose $d_k := \lceil (d+1)^{1/k} \rceil - 1$. By a similar logic as in the proof of Lemma 11, it is clear that $\psi_{k,d}$ will give distinct weights to distinct monomials. We have $\phi_{k,d_k} \circ \psi_{k,d} = \text{id}$ over the right $\mathbb{F}_{\leq d}[x]$, where id is the identity map. Thus, the map $\psi_{k,d}$, the right inverse of the map ϕ_{k,d_k}, is naturally called the *inverse Kronecker map*.

4.3.4 Are our polynomial families explicit ?

We are studying the measures defined in the previous chapter with focus on two polynomial families $f_d := (x+1)^d$ and $g_d := \sum_{i=0}^{d} 2^{i^2} x^i$. We now construct a multivariate family from each of these univariate family of polynomials and check explicitness for each.

First, we do this for the polynomial family g_d, for which explicitness is easier to argue. Taking cue from the procedure laid out is previous subsection, we apply the inverse Kronecker map. Consider the n-variate polynomial family $\{G_n(\overline{x})\}_n$ where $G_n(\overline{x}) = \psi_{n,d}(g_d)$, the inverse Kronecker map applied on g_d where $d := 2^n - 1$. Note from the construction in the previous subsection, that the maximum individual degree of any variable is bounded by $\lceil (d+1)^{1/n} \rceil - 1 = 1$. Thus, G_n is an n-variate multilinear polynomial.

Also, d is given to us in binary, which is similar to n being provided in unary. Thus, this setting is consisted with our earlier setting in which we defined **CH**-explicitness.

Lemma 12. $\{G_n(\overline{x})\}_n$ is **CH**-explicit.

Proof. Note that the coefficients of the polynomial have the form 2^{i^2} where $0 \leq i \leq 2^{n-1}$. We need to compute the i-th bit of the $j-th$ coeffiecient of G_n. Clearly, as the j-th coefficient is 2^{j^2}, every bit except the j^2-th bit is 0. Thus, in order to compute it's i-th bit b_i, we have the simple check :

$$b_i = \begin{cases} 1 & \text{if } i = j^2 \\ 0 & \text{otherwise} \end{cases}$$

Note that the bit-sizes of both i and j are bounded by $O(n)$. Thus, the above check takes at most $O(n \log n)$ time, using Karatsuba's multiplication algorithm. Hence, the family $\{G_n(\overline{x})\}_n$ is **CH**-explicit. □

Now, consider the family f_d and the corresponding multivariate polynomial family $\{P_n(\overline{x})\}_n$ where $P_n(\overline{x}) = \psi_{n,d}(f_d)$. The **CH**-explicitness of this family is much harder to argue. To do so, we use the following theorem, that was proved in [Bürgisser, 2009]. However, in [Bürgisser, 2009], all inputs were taken in binary. Thus the version we use is the unary form, restated in [Koiran, 2011].

Theorem 15. Let $p(n)$ be a polynomial and suppose $(a(n,k))_{n\in\mathbb{N},k\leq 2^{p(n)}}$ is **CH**-definable. Then, the following sum and product sequences $b(n)$ and $c(n)$ are **CH**-definable :

$$b(n) := \sum_{k=0}^{2^{p(n)}} a(n,k) \qquad c(n) := \prod_{k=0}^{2^{p(n)}} a(n,k)$$

We show that binomial coefficients are definable in **CH**. This proof is very similar to [Bürgisser, 2009, Cor 3.12] and has been demonstrated in [Dutta, Saxena, and Thierauf, 2020].

Theorem 16. Let $p(n)$ be a polynomial and $d_n \leq 2^{p(n)}$. Then, the sequence $a(n,i) = \binom{d_n}{i}$ is **CH**-definable.

Proof. Consider $f_{d_n} = (x+1)^{d_n} = \sum_{i=0}^{d_n} \binom{d_n}{i} x^i$. Plugging $x = 2^{d_n}$, we get :

$$f_{d_n}\left(2^{d_n}\right) = \sum_{i=0}^{d_n} \binom{d_n}{i} \cdot 2^{id_n}$$

Clearly $\binom{d_n}{i} < 2^{d_n}$. Hence, the bits of $\binom{d_n}{i}$ in the binary representation of $f_{d_n}\left(2^{d_n}\right)$ do not overlap for different i's and can therefore, be read off the bit representation of $f_{d_n}\left(2^{d_n}\right)$. Therefore, it is enough to show that $f_{d_n}\left(2^{d_n}\right)$ is **CH**-definable.

We now use Theorem 15. The only criteria left to verify is that $\left(2^{d_n}+1\right)$ is **CH**-definable. We have demonstrated how to show this in the proof of Lemma 12. Hence, we are done. □

4.4 Depth-4 Reduction and the connection with Valiant's Hypothesis

Let us start with a criterion proposed in by [Valiant, 1979] that puts a large class of polynomials in **VNP**.

Theorem 17. [Valiant's Criterion] Suppose $\phi : \{0,1\}^* \longrightarrow \mathbb{N}$ is a function in the class **#P**/Poly. Then the family (f_n) of polynomials defined by :

$$f_n := \sum_{e \in \{0,1\}^n} \phi(e) x_1^{e_1} x_2^{e_2} \cdots x_n^{e_n}$$

is in **VNP**.

In Section 4.1, we demonstrated the depth reduction of a circuit $\mathcal{C}$ to another circuit $\mathcal{C}'$ of size $O\left(s^3 d^6\right)$ of depth $O(\log d)$, where d is the degree of the polynomial computed by the circuit $\mathcal{C}$ and s is it's size. One of the important consequences of this result is that proving bounds for $O(\log d)$-depth circuits is sufficient to prove bounds for general circuits. Thus, while proving a lower bound, we may assume that we already have circuit in ***normal form***.

Further depth reduction results were obtained in [Agrawal and Vinay, 2008], [Koiran, 2012] and [Tavenas, 2015] and these give us a reduction to constant depth circuits, although with a slightly super-polynomial blow-up. Let us state the most relevant result below.

Theorem 18. [Depth-4 reduction] Let f be an n-variate degree d polynomial computed by a size s arithmetic circuit. Then, for any $0 < t \leq d$, f can be equivalently computed by a homogeneous $\Sigma\Pi\Sigma\Pi^t$ circuit of top-fan-in $s^{O(d/t)}$ and size $s^{O(t+d/t)}$.

We shall not use this theorem directly but the proof technique involved in achieving this result. Let us give a short gist of the proof technique :

- Consider the given circuit and flatten it to log-depth to get a circuit in normal form.
- Break the circuit into two parts : the first part is composed of the topmost t levels of multiplication gates and the addition gates above them and the second part is the rest bottom part of the circuit.
- By the properties of the universal circuit, the top part computes a polynomial of degree 5^t (imagine it isolated with the bottom part being replaced by new variables) in the polynomials computed in the bottom part. Consider it's dense representation.
- The degree of the bottom part is bounded by $\deg(f)/(3/2)^t$ and this too can be written as a sum of monomials in the dense form. In both this computation and above, use the fact that the numer of monomials in a polynomial of degree d in n variables is at most $\binom{n+d}{d}$.
- Optimization over the parameter t gives us the required result.

A key ingredient in the proof is the following lemma, due to [Fischer, 1994], which allows us to convert a monomial to an exponential sum of powers.

Lemma 13. [Fischer's trick] Let $\mathbb{F}$ be a field of characteristic 0 or $> r$. Any expression of the form $g = \sum_{i=1}^{k} \prod_{j=1}^{r} g_{ij}$ with $\deg\left(g_{ij}\right) \leq \delta$ can be written as $g = \sum_{i=1}^{k'} c_i g_i^r$ where $k' = k \cdot 2^r$, $\deg\left(g_i\right) \leq \delta$ and $c_i \in \mathbb{F}$. In particular, each $g_i \in \text{span}_{\mathbb{F}}\left(g_{i'j} \mid j\right)$ for some i'.

We will now prove the following theorem which establihes the connection between Conjecture 2 and Valiant's Hypothesis:

Theorem 19. Consider the family $g_d := \sum_{i=0}^{d} 2^{i^2} x^i$. If Conjecture 2 is true for g_d, then **VNP** is exponentially harder than **VP**.

Proof. The proof proceeds in two steps :

1. Construct the polynomial n-variate multilinear polynomial family $\{G_n(\overline{x})\}_n$ by applying the inverse Kronecker map $\psi_{n,d}$ on g_d, where $d = 2^n - 1$. The **CH**-explicitness of this family has already been shown earlier.
2. Prove that $\{G_n\}_{n\in\mathbb{N}} \in$ **VNP**.
3. Prove that if Conjecture 2 holds true, then $\text{size}(G_n) = d^{\Omega(1)} = 2^{\Omega(n)}$.

These steps exhibit a polynomial in **VNP** that has exponential circuit size, thereby showing that **VNP** is exponentially harder than **VP**.
Step 1 has already been executed. Let us start with Step 2.

Lemma 14. $\{G_n\}_{n\in\mathbb{N}} \in \mathbf{VNP}$.

Proof. Consider the following polynomial $\tilde{G_n}(\overline{x},\overline{y})$ in $3n$ variables $x_1, x_2, ..., x_n, y_1, y_2, ..., y_{2n}$:

$$\tilde{G_n}(\overline{x},\overline{y}) := \sum_{i=0}^{2^n-1} \overline{y}^{\text{bin}(i^2)} \cdot \overline{x}^{\text{bin}(i)}$$

where $\text{bin}(i) = (i_1, i_2, ..., i_n)$ is a vector such that $i = \sum_{j=1}^{n} i_j 2^{j-1}$. Substituting $y_i = 2^{2^{i-1}}$ $\forall$ $i \in [1, 2n]$ in $\tilde{G_n}$ gives G_n. In particular, we have :

$$\tilde{G_n}(\overline{x},\overline{y}) = \sum_{i=0}^{2^n-1} \sum_{j=0}^{2^{2n}-1} \phi(i,j)\, \overline{x}^{\text{bin}(i)} \cdot \overline{y}^{\text{bin}(j)}$$

where the function ϕ on $\mathbb{N}\times\mathbb{N}$ is defined as follows :

$$\phi(i,j) := \begin{cases} 1 & \text{if } j = i^2 \\ 0 & \text{otherwise} \end{cases}$$

The bit-size of the exponent vectors of $\overline{x}$ and $\overline{y}$ is $O(n)$. Using Karatsuba's multiplication algorithm, $\phi(i,j)$ can be computed in time $O(n \log n)$. Thus, $\phi \in \mathbf{FP} \subseteq \#\mathbf{P}/Poly$.

Hence, by Valiant's Criterion (Theorem 17), we get that $\left\{\tilde{G_n}\right\}_{n\in\mathbb{N}} \in \mathbf{VNP}$. Since, **VNP** is closed under substitution [Bürgisser, 2000], $\{G_n\}_{n\in\mathbb{N}}$ also belongs to **VNP**. □

We now prove the exponential hardness of the family $\{G_n\}_{n\in\mathbb{N}}$, assuming Conjecture 2 is true. From, now on, by the size of a circuit we will refer to the size of the equivalent universal circuit.

Lemma 15. Assuming Conjecture 2 is true, any *universal* circuit C computing $\{G_n\}_{n\in\mathbb{N}}$ has size at least $d^{\Omega(1)} = 2^{\Omega(n)}$.

Proof. We prove the claim by contradiction. Suppose *universal* circuit complexity of $\{G_n\}_{n\in\mathbb{N}}$ is not $d^{\Omega(1)}$. Then, $\exists$ an infinite domain $I \subset \mathbb{N}$ such that universal circuit complexity of $\{G_n\}_{n\in\mathbb{N}}$ is at most $d^{o(1)}$ over I. Tn particular, $\exists$ a function μ of d, such that size $(G_n) \leq d^{1/\mu(d)}$ and $\mu(d) \longrightarrow \infty$ over I. Without loss of generality, we can also assume that $\mu(d) \leq \log d$. Since G_n is an n-variate multilinear polynomial, we have $\deg(G_n) \leq n$.

We now look at the universal circuit $\mathcal{C}$ of size at most $s' := d^{1/\mu(d)}$ and cut the circuit at the t-th layer of multiplication gates from the top, where $t := t(d)$ is a function of d that we will fix later. We get the following two parts :

- **Top Part :** Since the fan-in of each multiplication gate is 5, the top part of the circuit computes a polynomial of degree at most 5^t. The number of variables is trivially bounded by s' , the size of the circuit. Hence, the top part can be written as a trivial $\Sigma\Pi$ circuit of size :

$$s_1'' := \binom{s' + 5^t}{5^t}$$

- **Bottom Part :** The number of multiplication gates that feed into the top part is bounded by s' . Since, $\deg(G_n) \leq n$, and the degree at least halves below every multiplication layer, the bottom part computes a polynomial of degree at most $n \cdot 2^{-t}$. The number of variables is n. So, each multiplication gate at the top of the bottom layer can be reduced to a trivial $\Sigma\Pi$ circuit of size :

$$s_2'' := \binom{n + n2^{-t}}{n}$$

From the above construction, we have a $\Sigma^{s_1''}\Pi^{5^t}\Sigma^{s_2''}\Pi^n$ circuit computing G_n. In particular, using Fischer's trick (Lemma 13), one can express :

$$G_n = \sum_{i=1}^{s_1''} \prod_{j=1}^{5^t} g_{ij} = \sum_{i=1}^{s_1'' \cdot 2^{5^t}} c_i g_i^{5^t}$$

where $g_i \in \text{span}_{\mathbb{F}}\left(g_{i'j} | j\right)$ for some i' and $c_i \in \mathbb{F}$. Each g_{ij} is an n-variate polynomial of degree at most $n \cdot 2^{-t}$. Thus, each g_i is an n-variate polynomial of degree at most $n \cdot 2^{-t}$.

Choose $r := r(d)$ such that r is a prime and $5^t < r < 5^{t+1}$. We know that such a prime exists because of Bertrand's Postulate. Using Theorem 8 , we know that there exists $c_{ij}, \lambda_j \in \mathbb{F}$ such that :

$$g_i^{5^t} = \sum_{i=1}^{r+1} c_{ij}\left(g_i + \lambda_j\right)^r$$

In particular, $\exists\ \tilde{c}_i \in \mathbb{F}$, such that :

$$G_n = \sum_{i=1}^{\tilde{s}} \tilde{c}_i \tilde{g}_i^{\,r}$$

where $\tilde{s} := s_1'' \cdot 2^{5^t} \cdot (r+1)$ and each $\tilde{g}_i$ is an n-variate polynomial of degree at most $n \cdot 2^{-t}$.

Now applying the Kronecker map $\phi_{n,1}$ to G_n yields :

$$g_d := \phi_{n,1}(G_n) := \sum_{i=1}^{\tilde{s}} \tilde{c}_i \cdot \phi_{n,1}\left(\tilde{g}_i\right)^r$$

An important observation is that $\left|\bigcup_i \text{supp}\left(\tilde{g}_i\right)\right| \leq s_2'' \implies \left|\bigcup_i \text{supp}\left(\phi_{n,1}\left(\tilde{g}_i\right)\right)\right| \leq s_2''$.
Thus, we must have :

$$U_{\mathbb{F}}\left(g_d, r, \tilde{s}\right) \leq s_2''.$$

So, once t is fixed, we have that $\tilde{s}$ and r gets fixed (up to constant multiple). We need to fix a t such that $\tilde{s} \leq d^{o(1)}$, $r \leq \log^* d$-th prime $s_2'' < o\left(d/r\right)$. If such t exists, then we are done because that shows $U_{\mathbb{F}}\left(g_d, r, \tilde{s}\right) \leq o\left(d/r\right)$ over an infinite domain I, which would be a contradiction.

Let

$$5^t \;:=\; \min\left(\sqrt{\mu(d)}, (1/5)\cdot p_{\log^* d}\right)$$

where $p_{\log^* d}$ denotes the $\log^* d$-th prime. As $r < 5^{t+1}$, hence $r \leq \log^* d$-th prime, by definition. Also, $r \geq 5^t$, and so $r(d) \longrightarrow \infty$ as $d \longrightarrow \infty$. Also, notice that $t(d) \longrightarrow \infty$ as $\mu(d) \longrightarrow \infty$.

Note that the following result holds true.

$$\text{For } n \geq d \geq 0, \text{ we have } \binom{n+d}{d} \leq \min\left(n^{O(d)}, d^{O(n)}\right)$$

This gives us :

$$\tilde{s} = s_1'' \cdot 2^{5^t} \cdot (r+1) \;\leq\; \left(s'\right)^{O\left(5^t\right)} \cdot 5^t \;\leq\; d^{O\left(1/\sqrt{\mu(d)}\right)} \cdot \sqrt{\mu(d)} \;\leq\; d^{o(1)}$$

What we have left to show is that $s_2'' \leq o(d/r)$. We have :

$$s_2'' \;=\; \binom{n+n2^{-t}}{n} \;\leq\; \left(e\left(1+2^t\right)\right)^{n2^{-t}} \;\leq\; \left(\frac{7}{2}\cdot 2^t\right)^{n2^{-t}}$$

On the other hand $\dfrac{d}{r} \geq \dfrac{(2^n-1)}{p_{\log^* d}}$. Note that :

$$\frac{s_2''}{\frac{d}{r}} \;\leq\; \frac{2^{n/(t+3)}}{2^{n-3t}} \cdot \left(\frac{5}{8}\right)^t$$

Notice $t \;:=\; o(\log d)$ and $n = O\left(\log d\right)$ and both functions $\to \infty$ as $d \to \infty$. From these observations and the above inequality, it is obvious that :

$$\lim_{d\to\infty}\left(\frac{s_2''}{\frac{d}{r}}\right) \longrightarrow 0 \qquad \text{as } d \longrightarrow \infty$$

Thus, $s_2'' = o(d/r)$. Thus, the universal circuit complexity of G_n is at least $d^{\Omega(1)} = 2^{\Omega(n)}$ □

This completes the proof of Theorem 19. Note that while we have done this for the polynomial family g_d, the proof of hardness is exactly the same for the family $f_d = (x+1)^d$. The only catch is in the proof of inclusion of the n-variate multilinear polynomial family $P_n = \psi_{n,1}\left(f_d\right)$ in **VNP**. We have shown that the coefficient sequence of binomials is **CH**-definable. Thus, by Valiant's Criterion, for $\{P_n\}_{n\in\mathbb{N}}$, we would achieve exponential separation of **VNP** from **VP** (under Conjecture 2), if **CH** = **#P**/Poly. Thus, we have also proved the following result which is very much in the spirit of the famous *derandomization* $\implies$ *hardness* result of Kabanets-Impagliazzo [Kabanets and Impagliazzo, 2003].

Theorem 20. Consider the family $f_d := (x+1)^d$. If Conjecture 2 is true for f_d, then either **CH** $\neq$ **#P**/Poly or **VNP** is exponentially harder than **VP**.

□

Chapter 5

The Sum of Squares Model and Counterexample Generation

We now focus our attention to a special case of the SOP model : the *Sum of Squares* model for the family f_d , which is simply the following :

$$(x+1)^d = \sum_{i=1}^{s} l_i^2$$

Also, for this chapter, we focus our attention on the measure $S_{\mathbb{F}}$ i.e the minimum of the sum of sparsity of the constituent representative polynomials and try to attempt to prove the ambitious conjecture that :

$$S_{\mathbb{F}}\left((x+1)^d, 2, s\right) \geq \Omega(d)$$

We will mostly outline some natural attempts to prove this conjecture, outline why they don't work and come up with a systematic measure to generate some striking univariate polynomial identities in the process.

We have mostly worked with two specific polynomial families throughout this work, stating their importance in their **explicitness**. Clearly these might not be the only choices . In the first section, we speak of some choices that work and for proof, point to [Dutta, Saxena, and Thierauf, 2020]. Also, we point out some choices that don't and for a proof we revisit the idea of *sumsets*. Roughly, the idea is to be able to write a dense polynomial as a product of two sparse polynomials and then use the simple *product-to-sum of squares* conversion.

5.1 Some useful and some refutable choices of polynomials

In this section, we speak about our choice of polynomial. We have seen until now, that we have mostly prefered working with the polynomial families : $f_d = (x+1)^d$ and $g_d = \sum_{i=0}^{d} 2^{i^2} x^i$. Are these the only candidate polynomial families ? Of course not. Very similar polynomial families work as well as they too fit the criterion for **CH**-explicitness. Consider $h_d := \sum_{i=0}^{d} 3^{i^2} x^i$ or $q_d := \prod_{i=1}^{d} (x+i)$. To see how the family q_d is **CH**-explicit, consult [Dutta, Saxena, and Thierauf, 2020]. The proof is along the same lines as the proof of explicitness of f_d.

Let us now look at a candidate polynomial family which does not work. Consider

$$p_d := \sum_{i=0}^{d} x^i$$

To see why this polynomial does not work, recall the idea of *sumsets*. We know that :

$$|A+B| \leq |A| \cdot |B|$$

The key to refute this polynomial family is to find sets $|A|$ and $|B|$ such that each is of size $\sqrt{d}$ and the equality in the above equation is satisfied. Without loss of generality, let us assume d is a perfect square (since perfect squares form an infinite domain, showing that the lower bound is violated for this domain is enough to refute the conjecture) . So, consider the following two sets :

$$A := \left\{a |\ a \in [\sqrt{d}-1]\right\}$$

$$B := \left\{b\sqrt{d} |\ b \in [\sqrt{d}]\right\} \tag{5.1}$$

Notice that each element $x \in [d]$ can be written in $\sqrt{d}$-basis as $a + b\sqrt{d}$ where $a \in [\sqrt{d}-1]$ and $b \in [\sqrt{d}]$. Thus, we have :

$$[d] := A + B$$

which gives us :

$$\sum_{1=0}^{d} x^i := \left(\sum_{i \epsilon A} x^i\right) \cdot \left(\sum_{i \epsilon B} x^i\right)$$

Now, using $l_1 l_2 = (l_1+l_2)^2 + i^2(l_1-l_2)^2$, we are able to write :

$$\sum_{1=0}^{d} x^i := (l_1+l_2)^2 + i^2(l_1-l_2)^2$$

where $l_1 = \sum\limits_{i \epsilon A} x^i$ and $l_2 = \sum\limits_{i \epsilon B} x^i$, both of which have sparsity $O\left(\sqrt{d}\right)$. Thus, this refutes the conjecture for this particular family. It is therfore obvious that any family like $\sum\limits_{i=0}^{d} c^i x^i$ won't work either.

Consider the special case :

$$(x+1)^{2d} - l_1^2 = l_2^2 - l_3^2$$

Considering this format makes sense when we are working over $\mathbb{C}$. Note that, using $a^2 - b^2 = (a+b)(a-b)$, one can ask the general question if the product of two sparse polynomials can be written as the product of two dense polynomials. If this is not possible, it will resolve this special case of three summands. Note that we have already seen that a dense polynomial can be written as the product of two sparse polynomials. Now consider the following example :

$$\left(x^d - 1\right)\left(2^d x^d - 1\right) = (x-1)\left(\sum_{i=0}^{d-1} x^i\right) \cdot (2x-1)\left(\sum_{i=0}^{d-1} 2^i x^i\right)$$

Let $f = (x-1)\left(\sum_{i=0}^{d-1} 2^i x^i\right)$ and $g = (2x-1)\left(\sum_{i=0}^{d-1} x^i\right)$. Note that both f and g have sparsity d, while each term in the L.H.S has sparsity 2. Thus, the product of two dense polynomials can be written as the product of two sparse polynomials. Thus, even this case of three squares remains open.

Another possible attempt could be the following conjecture :

$$\sum_i l_i^2 = f^2 \implies f \in \text{span}_{\mathbb{F}}\{l_1, l_2, ..., l_s\}$$

Notice that this would immediate imply the conjecture for our candidate polynomial families as our measure is sub-additive. However, this conjecture is immediately refuted by the following counterexample :

$$l_1 = a^2 + b^2 - c^2, \quad l_2 = 2ac, \quad l_3 = 2bc. \text{ and } \quad l_4 = a^2 + b^2 + c^2$$

Notice that : $l_1^2 + l_2^2 + l_3^2 = l_4^2$ but $l_4 \notin \text{span}_{\mathbb{F}}\{l_1, l_2, l_3\}$.

However, while this conjecture is not true for general f , it is not immediate while such a conjecture should not be true for the monomial $f = x^d$. Notice that, if this is true, it would imply our conjecture since we can just use the shift map :

$$x \longrightarrow x + 1$$

and the conjecture would follow from the spanning criterion. However, this conjecture fails as well. For $a \neq d$, write :

$$x^{2d} = x^{2d-a} \cdot x^a$$

Using this and technique demonstrated earlier for writing a product as a sum of squares, we can write :

$$x^{2d-a} \cdot x^a + x^b \cdot x^{2d-b} := \sum_{i \in [3]} l_i^2$$

which clearly refutes the conjecture.

However, a more viable conjecture might be :

$$x^d \cdot g(x) \in \text{span}_{\mathbb{F}}\{l_1, l_2, ..., l_s\} \tag{5.2}$$

for some non-zero polynomial g. The following lemma shows why this should imply the conjecture.

Lemma 16. [Hajós Lemma] Suppose $f(x) \in \mathbb{C}[x]$ be a univariate polynomial with $t \geq 1$ monomials. Let α be a non-zero root of $f(x)$. Then, the multiplicity of α in f can be at most $t-1$.

For the sake of completion, we include a proof of the above lemma.

Proof. The proof will be by induction on t. For $t = 1$, $f(x) = a_k x^k$, which has no non-zero roots. Hence, we are trivially done. For $t \geq 2$, assume that $f(x) = x^r \cdot g(x)$ for some r such that sparsity$(g) = t$ and $g(0) \neq 0$. Notice that it is enough to prove the result for g as multiplication by x^r only shifts the exponents and does not change sparsity. Also, all non-zero roots of f are non-zero roots of g and vice-versa. Let α be a root of g. Now, sparsity $(g') = t - 1$ as $g(0) \neq 0$. Thus, by induction hypothesis, multiplicity of α in g is at most $t - 2$. This implies that the multiplicity of α in f is at most $t - 1$. □

The above lemma shows that sparsity $\left((x+1)^d \cdot g(x)\right) \geq d + 1$ for any non-zero polynomial g . Thus, if (2) holds, then we can apply the shift map and reach the desired conclusion.

5.2 Counterexample generation

In the earlier section, we had refuted the conjecture that :

$$x^{2d} = \sum_i l_i^2 \implies x^d \in \text{span}_{\mathbb{F}} \{l_1, l_2, ..., l_s\}$$

However, our counterexample had a very special property : for each l_i there was a nonzero monomial in l_i with degree $> d$. This leads to the following question : Suppose we restricted ourselves only to those l_i which have maximum degree d and studied the representation :

$$x^{2d} = \sum_i c_i l_i^2$$

then what can we say about the l_i's ? Of course we would also like to bound the number of l_i's. The most natural condition to put is to consider that we have $< d$ l_i's, otherwise the sparsity would already be too large. For the rest of this chapter, we fix $\mathbb{F} = \mathbb{C}$. Also, we simplify the above representation, by using the *Hadamard Product of Vectors*.

Definition 20. For two matrices A and B of the same dimension $m \times n$, the Hadamard product $A \circ B$ is a matrix of the same dimension as the operands, with elements given by :

$$(A \circ B)_{ij} := (A)_{ij} B_{ij}$$

For matrix of different dimensions, the Hadamard product is undefined.

For our purpose the Hadamard product of two vectors u and v of the same dimension, will be denoted simply as uv. In case we wish to indicate the *inner product* of u and v, we will denote it as $u^T v$, where u^T denotes the transpose of u.

So, we have to examine an identity of the form :

$$x^{2d} = \overline{\alpha}^T \left(u_0 + u_1 x + u_2 x^2 + \cdots + u_d x^d\right)^2 \tag{5.3}$$

where the j-th co-ordinate of u_i is simply the coefficient of x^i in l_j and u_i's $\in \mathbb{C}^k$, where $k < d$.

We have a set of vectors u_i, $i \in [d]$. The most natural course of action is to first begin by considering the linear relations among the u_i's. Suppose $\{u_1, u_2, ..., u_k\}$ forms a basis for $\mathbb{C}^k$. We first emphasize that it is enough to consider $k = o(d)$. Suppose not and $k = \Omega(d)$. Without loss of generality, let us assume that u_d must be included in the basis set. Let the entire basis set be, again, without loss of generality, $\{u_1, u_2, u_3, ..., u_{k-2}, u_d\}$. Since, $\dim\langle u_1, u_2, ..., u_{k-2}\rangle_{\mathbb{C}} = k-1$, $\exists\ c \in \mathbb{C}^k$ such that $c^T \cdot u_i = 0$ $\forall\ i \leq k-2$ but $c^T \cdot u_d \neq 0$. Thus, we get $x^d = \sum c_i l_i$. By shifting, we can therefore prove our original sparsity conjecture. Thus, the case $k = \Omega(d)$ is uninteresting for us. Also, note that this is in sync with our original goal of avoiding examples with ***large monomials*** in the linear span.

Notice that only the monomial x^{2d} survives after expanding out the R.H.S of (3). So, there are a huge number of cancellations that take place. Since, we have restricted the degrees to be $\leq d$, it is not unnatural to wonder if so many cancellations would imply that a large number of u_i's must be non-xero. In spirit of enquiry, we make the following ambitious conjecture :

Conjecture 3. Suppose (3) holds. Also, suppose we have the following condition :

- $\{u_1, u_2, ..., u_k\}$ form a spanning set for $\{u_1, u_2, ..., u_d\}$ where $k = o(d)$

Then $\Omega(d)$ of the u_i's are non-zero.

5.2.1 The case $k = 3$, $d = 3$

For the rest of this write-up, we'll be working with the Hadamard product of vectors. We will first deal with this simple case where we are looking at a representation of the following form :

$$x^6 = \overline{\alpha}^T \left(1 + u_1 x + u_2 x^2 + u_3 x^3\right)^2 \tag{5.4}$$

where 1, u_1 and u_2 form a basis for $\mathbb{C}^3$. We therefore have two equations :

$$\overline{\alpha}^T \cdot [1, 2u_1, 2u_2 + u_1^2, 2u_3 + 2u_1 u_2] = 0$$

and

$$\overline{\alpha}^T \cdot [2u_1 u_3 + u_2^2, 2u_2 u_3] = 0$$

Also, we need : $\overline{\alpha}^T \cdot u_3^2 \neq 0$. The first set of equations will be termed as *variable setting* equations and the next set as the *constraint* equations. Our strategy to satisfy both equations, will be as follows :

- Pick $\overline{\alpha}$ to be orthogonal to 1 and u_1.
- Ensure that all the individual vectors within the [] in both sets, lie in the space spanned by 1 and u_1.

If the above two conditions are satisfied, then both sets of equations are automatically satisfied. Thus, we now have the following set of equations to satisfy :

$$2\begin{cases} u_1^2 + 2u_2 = 0 (\text{mod } V) \\ u_1u_2 + u_3 = 0 (\text{mod } V) \\ u_2^2 + 2u_1u_3 = 0 (\text{mod } V) \\ u_2u_3 = 0 (\text{mod } V) \\ u_3^2 \neq 0 (\text{mod } V) \end{cases}$$

where V is the subspace spanned by the vectors 1 and u_1. These equations may be rewritten as :

$$\begin{cases} u_2 = \frac{-u_1^2}{2} + \alpha_1 u_1 + \alpha_0 \\ u_3 = \frac{u_1^3}{2} - \alpha_1 u_1^2 + \beta_1 u_1 + \beta_0 \\ u_2^2 + 2u_1u_3 = 0 (\text{mod } V) \\ u_2u_3 = 0 (\text{mod } V) \\ u_3^2 \neq 0 (\text{mod } V) \end{cases}$$

The constraint equations can be rewritten in terms of just u_1 as follows :

$$\frac{3}{4}u_1^4 - 2\alpha_1 u_1^3 + \left(\alpha_1^2 + \beta_1 - \alpha_0\right) u_1^2 = 0 (\text{mod } V)$$

and

$$\frac{-1}{4}u_1^5 + \alpha_1 u_1^4 + \left(\frac{\alpha_0}{2} - \frac{\beta_1}{2} - \alpha_1^2\right) u_1^3 + \left(\alpha_1\beta_1 - \alpha_0\alpha_1 - \frac{\beta_0}{2}\right) u_1^2 = 0 (\text{mod } V)$$

To convert it from an equation concerning vectors to linear equations over complex numbers, we first fix u_1. From the expression of u_2 in terms of u_1 and the fact that 1, u_1 and u_2 form a basis, it is simple to see that $1, u_1, u_1^2$ form a basis too. Thus, by fixing this basis, a vector w modulo V, is simply the coefficient c_2 when w is expressed as $w = c_0 + c_1u_1 + c_2u_1^2$.

Fix $u_1 = [-1, 0, 1]$. It is simple to see that $u_1^k = 1 (\text{mod } V)$ for any even $k \in \mathbb{N}$ and $u_1^k = 0 (\text{mod } V)$ for any odd $k \in \mathbb{N}$. We also fix $(\alpha_0, \alpha_1) = (0, 0)$.

On solving the linear equations obtained, we get $\beta_1 = \frac{-3}{4}, \beta_0 = 0$. Using these, it is also simple to check that $u_3^2 \neq 0 (\text{mod } V)$. This gives us the following counterexample :

$$l_1 = \left(1 - x - \frac{1}{2}x^2 + \frac{1}{8}x^3\right)$$

$$l_2 = 1$$

$$l_3 = \left(1 + x - \frac{1}{2}x^2 - \frac{1}{8}x^3\right)$$

and $\overline{\alpha} = 32[1, -2, 1]$.

5.2.2 The algorithm for general d and k

We turn ourselves to the case of general d and k, i.e we have $u_1, u_2, u_3, ..., u_k$ as the basis set. We claim that it is sufficient to restrict ourselves to the case where $u_1 = 1$

i.e the following :

$$x^{2d} = \overline{\alpha}^T \cdot \left(1 + u_1 x + u_2 x^2 + ... + u_d x^d\right)^2$$

Suppose not. Without loss of generality, we have $x^{2d} = \sum \alpha_i l_i^2$ and $l_1 = x + \sum_{i>1} c_i x^i$. Then, we have $l_1^2 = x^2 \left(1 + \sum_{i>2} c_i x^{i-1}\right)^2$. Then, we use the identity $A^2B^2 = \left(\frac{A^2+B^2}{2}\right)^2 + i^2\left(\frac{A^2-B^2}{2}\right)^2$. Since, $A = x$ and $B = \left(1 + \sum_{i>2} c_i x^{i-1}\right)$, we reduce to the case $u_1 = 1$. So, we are done.

We now consider an equivalent problem. Consider the expression :

$$E(x) = \left(1 + u_1 x + u_2 x^2 + ... + u_d x^d\right)^2$$

We would like to ask the following structural question about $E(x)$.

Conjecture 4. Suppose the following holds true about E :

- Generator set of $\{1, u_1, u_2, ..., u_d\}$ is up to $u_{k=o(d)}$
- $\text{coeff}_{x^{2d}}(E(x)) \notin \langle \text{coeff}_{x^i}(E(x)) | 0 \le i < 2d \rangle_{\mathbb{C}}$

Then, $\Omega(d)$ of the u_i's are nonzero.

We would like to demonstrate that there exists counterexamples for this conjecture. In fact, we would like to construct an algorithm that, under favorable choices of initial parameters, generates such counterexamples. The idea to construct counterexamples is similar to the example presented in the previous section.

Theorem 21. There exists a randomized algorithm A, which, on input of an orthonormal basis $V = \{v_1, v_2, ..., v_k\}$ for $\mathbb{C}^k$ containing the all 1 vector 1 either outputs a counterexample to Conjecture 4 with $u_1 = v_1$ or outputs *failure* after a finite number of checks.

Proof. Let the basis set be $\{v_1, v_2, ..., v_k\}$, with $v_1 = 1$ and $v_2 = u_1$. We let $V = \langle v_1, v_2, ..., v_{k-1} \rangle_{\mathbb{C}}$. Our strategy would be to consider a counterexample of the form :

$$E(x) = \left(1 + u_1 x + u_2 x^2 + ... + u_k x^k + u_d x^d\right)^2$$

So, what we would like to have is $\langle \text{coeff}_{x^i}(E(x)) | 0 \le i < 2d \rangle_{\mathbb{C}} \subseteq V$ and pick u_d^2 from the subspace spanned by v_k. This would immediately satisfy both the criteria and yet be a counterexample since only $k + 1 = o(d)$ of the u_i's are non-zero. Thus, we would like to solve the following system :

$$coeff_{x^i}\left(E(x)\right) = 0 (\text{mod } V)$$

for $0 \leq i \leq 2d-1$. We will first obtain a set of variable setting equations and then a set of constraint equations just as in the previous example. The algorithm that we give simply ensures fixing of certain variables in a proper fashion so that at every stage, we only have systems of linear equations.

The algorithm proceeds as follows :

- Take as input the basis $\{v_1, v_2, ..., v_k\}$ for $\mathbb{C}^k$. Also, fix u_1.
- Consider the subspace $V = \langle v_1, v_2, ..., v_{k-1} \rangle$ and $v_1 = 1$ and $v_2 = u_1$.
- For $i = 2$ to k, do :
  - Consider the coefficient of x^i in $E(x)$. We have :

    $$coeff_{x^k}(E(x)) = 0 (\text{mod } V)$$

    which may be rewritten as ;

    $$u_i = \sum_{r+s=i, rs \neq 0} u_r u_s + \sum_{j=1}^{k-1} \alpha_{ij} v_j$$

    where α_{ij} are some yet unfixed variables. In the case, where $s > r$ in the above convolution, randomly fix u_r (by randomly fixing it's corresponding α's), but keep u_s unfixed. The invariant that we shall maintain in this part of the algorithm (and even consequently) is that every u_i is defined as a linear polynomial in the α's. Proceed inductively. On the R.H.S $r, s < i$ in every term of the convolution. Hence, when u_r is fixed, u_s is linear in its α's (by Induction hypothesis. The base cases are u_1 and u_2, which are easily checked.). Thus, we conclude that u_i is linear in α_{ij}'s.
- At the end of the above procedure, we have fixed all the u_i's with $i \leq \frac{k}{2}$. The remaining remain unfixed.
- We now move to the constraint equations. We assume that we posses a subroutine that allows us to compute any $v_i^m v_j^n$ modulo V. For $i = k+1$ to $2k$, do :
  - Consider the coefficient of x^i. We have :

    $$\sum_{r+s=i} u_r u_s = 0 (\text{mod } V)$$

    We now proceed as earlier. Whenever $s > r$, randomly fix u_r (if it was already fixed during the variable setting procedure, then consider that value which was fixed during that procedure.) but keep u_s unfixed. The same invariant is maintained. By using the aforementioned subroutine, the subspace inclusion constraint is transformed in to an equation over the α's of the unfixed u_s's, which, by the invariant property, is a linear equation.

  By this fixing procedure, at the last step i.e $i = 2k$, we have to fix the last remaining generating vector i.e u_k. Thus, after the end of this procedure, all $u_1, u_2, ..., u_k$ are fixed. Also, we note that u_d does not appear in any of the convolution sums since $k = o(d)$.

- Since, u_d^2 is picked from the one-dimensional subspace orthogonal to V, it is fixed upto a scalar. So, u_d is fixed upto a scalar. We finally have to check the constraints ;

$$u_d u_i = 0 (\text{mod } V)$$

for $1 \leq i \leq k$. Note that since we are equating to 0, the scaling factor for u_d does not matter. So, we can simply take $u_d = \sqrt{v_k}$. If any of these checks fail, return *Failure*, otherwise return $u_1, u_2, ..., u_k$.

This concludes the description of the counterexample generating algorithm. □

Chapter 6

Conclusion

In conclusion, we looked at lower bounds for univariate polynomials using sparsity based measures. We introduced two sparsity based measures and proved lower bounds for explicit polynomial families for certain specific parameters and an unconditional lower bound for localized integer rings. We formulated a conjecture saying that the Sum of Powers representation for certain specific polynomial families should be large in terms of the sparsity measures. We showed that proving this conjecture would imply lower bounds, specifically, Valiant's Hypothesis. We looked at the special case of sum of Squares representation and showed certain plausible approaches that don't work. As a result, we found a systematic approach for proving striking polynomial identities. However, certain questions remain open.

- It was shown in [Dutta, Saxena, and Thierauf, 2020] that if $(x+1)^d$ written as a sum of $o(d)$ many 25-th powers of univariates requires $\Omega(d)$ many distinct monomials, then we can derandomize blackbox PIT as well as prove Valiant's hypothesis. This proof crucially uses the hardness to efficient derandomization results of [Guo et al., 2019] and [Agrawal, Ghosh, and Saxena, 2018]. [Dutta and Saxena, 2020] furthered improved this by showing that studying 4-th powers is enough to achieve the above results. Their proof uses a novel depth-reduction idea instead of the standard [Valiant et al., 1983] that gives a new *normal form* circuit. We ask if this can be further reduced to the case of squares, which is the simplest model to study.

- Prove Conjecture 2. In fact, prove the slightly differently formulated conjecture for 4-th powers in [Dutta and Saxena, 2020] .

- Consider the result established in [Garca-Marco and Koiran, 2017] for powers of linear forms. Improve this to powers of quadratic forms. This might be easier than the case of sum of squares since the exact form of the polynomial that is being powered is known in this case.

- Prove that the measure $U_{\mathbb{F}}(\cdot)$ is *large* for *random* polynomials.

- Show if simpler lower bounds such as $\Omega\left(\sqrt{d}\right)$ suffice to reach the conclusion $\mathbf{VP} \neq \mathbf{VNP}$.

Bibliography

Agrawal, M. and V. Vinay (2008). "Arithmetic Circuits: A Chasm at Depth Four". In: *2008 49th Annual IEEE Symposium on Foundations of Computer Science*, pp. 67–75.

Agrawal, Manindra, Sumanta Ghosh, and Nitin Saxena (2018). "Bootstrapping variables in algebraic circuits". In: *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*. Ed. by Ilias Diakonikolas, David Kempe, and Monika Henzinger. ACM, pp. 1166–1179. DOI: 10.1145/3188745.3188762. URL: https://doi.org/10.1145/3188745.3188762.

Allender, Eric et al. (1998). "Non-Commutative Arithmetic Circuits: Depth Reduction and Size Lower Bounds". In: *Theor. Comput. Sci.* 209.1-2, pp. 47–86. DOI: 10.1016/S0304-3975(97)00227-2. URL: https://doi.org/10.1016/S0304-3975(97)00227-2.

Baur, Walter and Volker Strassen (1983). "The Complexity of Partial Derivatives". In: *Theor. Comput. Sci.* 22, pp. 317–330.

Brent, Richard P. (Apr. 1974). "The Parallel Evaluation of General Arithmetic Expressions". In: *J. ACM* 21.2, 201–206. ISSN: 0004-5411. DOI: 10.1145/321812.321815. URL: https://doi.org/10.1145/321812.321815.

Bürgisser, Peter (2000). *Completeness and Reduction in Algebraic Complexity Theory*. Vol. 7. Algorithms and computation in mathematics. Springer. ISBN: 978-3-540-66752-0.

— (2009). "On Defining Integers And Proving Arithmetic Circuit Lower Bounds". In: *Comput. Complex.* 18.1, pp. 81–103. DOI: 10.1007/s00037-009-0260-x. URL: https://doi.org/10.1007/s00037-009-0260-x.

Bürgisser, Peter, Michael Clausen, and Amin Shokrollahi (Jan. 1997). *Algebraic Complexity Theory*. Vol. 315. DOI: 10.1007/978-3-662-03338-8.

Chatterjee, Prerona et al. (2019). *A Quadratic Lower Bound for Algebraic Branching Programs and Formulas*. arXiv: 1911.11793 [cs.CC].

Chen, Xi, Neeraj Kayal, and Avi Wigderson (2011). "Partial Derivatives in Arithmetic Complexity and Beyond". In: *Foundations and Trends® in Theoretical Computer Science* 6.1–2, pp. 1–138. ISSN: 1551-305X. DOI: 10.1561/0400000043. URL: http://dx.doi.org/10.1561/0400000043.

Dutta, Pranjal and Nitin Saxena (2020). "Lower-bounding the sum of 4th-powers of univariates leads to derandomization and hardness". In: URL: https://www.cse.iitk.ac.in/users/nitin/papers/4th-powers.pdf.

Dutta, Pranjal, Nitin Saxena, and Thomas Thierauf (2020). "Lower bounds on the sum of 25th-powers of univariates lead to complete derandomization of PIT". In: *Electronic Colloquium on Computational Complexity (ECCC)* 27, p. 39. URL: https://eccc.weizmann.ac.il/report/2020/039.

Fine, N. J. (1947). "Binomial Coefficients Modulo a Prime". In: *The American Mathematical Monthly* 54.10, 589—592. ISSN: 00029890, 19300972. URL: http://www.jstor.org/stable/2304500.

Fischer, Ismor (1994). "Sums of Like Powers of Multivariate Linear Forms". In: *Mathematics Magazine* 67.1, pp. 59–61. DOI: 10.1080/0025570X.1994.11996185. eprint: https://doi.org/10.1080/0025570X.1994.11996185. URL: https://doi.org/10.1080/0025570X.1994.11996185.

Garca-Marco, Ignacio and Pascal Koiran (Apr. 2017). "Lower Bounds by Birkhoff Interpolation". In: *J. Complex.* 39.C, 38–50. ISSN: 0885-064X. DOI: 10.1016/j.jco.2016.10.001. URL: https://doi.org/10.1016/j.jco.2016.10.001.

Grigoriev, Dima and Marek Karpinski (1998). "An Exponential Lower Bound for Depth 3 Arithmetic Circuits". In: *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*. STOC '98. Dallas, Texas, USA: Association for Computing Machinery, 577–582. ISBN: 0897919629. DOI: 10.1145/276698.276872. URL: https://doi.org/10.1145/276698.276872.

Guo, Zeyu et al. (2019). "Derandomization from Algebraic Hardness: Treading the Borders". In: *60th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2019, Baltimore, Maryland, USA, November 9-12, 2019*. Ed. by David Zuckerman. IEEE Computer Society, pp. 147–157. DOI: 10.1109/FOCS.2019.00018. URL: https://doi.org/10.1109/FOCS.2019.00018.

Gupta, Ankit et al. (Dec. 2014). "Approaching the Chasm at Depth Four". In: *J. ACM* 61.6. ISSN: 0004-5411. DOI: 10.1145/2629541. URL: https://doi.org/10.1145/2629541.

— (2016). "Arithmetic Circuits: A Chasm at Depth 3". In: *SIAM J. Comput.* 45.3, pp. 1064–1079. DOI: 10.1137/140957123. URL: https://doi.org/10.1137/140957123.

Kabanets, Valentine and Russell Impagliazzo (2003). "Derandomizing Polynomial Identity Tests Means Proving Circuit Lower Bounds". In: *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing*. STOC '03. San Diego, CA, USA: Association for Computing Machinery, 355–364. ISBN: 1581136749. DOI: 10.1145/780542.780595. URL: https://doi.org/10.1145/780542.780595.

Kalorkoti, K. A. (Aug. 1985). "A Lower Bound for the Formula Size of Rational Functions". English. In: *SIAM Journal on Computing* 14.3, pp. 678–687. ISSN: 0097-5397. DOI: 10.1137/0214050.

Kayal, Neeraj and Ramprasad Saptharishi (2014). "A Selection of Lower Bounds for Arithmetic Circuits". In: *Perspectives in Computational Complexity: The Somenath Biswas Anniversary Volume*. Ed. by Manindra Agrawal and Vikraman Arvind. Cham: Springer International Publishing, pp. 77–115. ISBN: 978-3-319-05446-9. DOI: 10.1007/978-3-319-05446-9_5. URL: https://doi.org/10.1007/978-3-319-05446-9_5.

Kayal, Neeraj et al. (2015). "Lower Bounds for Sums of Powers of Low Degree Univariates". In: *Automata, Languages, and Programming*. Ed. by Magnús M. Halldórsson et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 810–821. ISBN: 978-3-662-47672-7.

Koiran, Pascal (2011). "Shallow circuits with high-powered inputs". In: *Innovations in Computer Science - ICS 2011, Tsinghua University, Beijing, China, January 7-9, 2011. Proceedings*. Ed. by Bernard Chazelle. Tsinghua University Press, pp. 309–320. URL: http://conference.iiis.tsinghua.edu.cn/ICS2011/content/papers/5.html.

— (2012). "Arithmetic circuits: The chasm at depth four gets wider". In: *Theor. Comput. Sci.* 448, pp. 56–65. DOI: 10.1016/j.tcs.2012.03.041. URL: https://doi.org/10.1016/j.tcs.2012.03.041.

Lovett, Shachar (2011). "Computing Polynomials with Few Multiplications". In: *Theory of Computing* 7.13, pp. 185–188. DOI: 10.4086/toc.2011.v007a013. URL: http://www.theoryofcomputing.org/articles/v007a013.

— (2017). "Additive Combinatorics and its Applications in Theoretical Computer Science". In: *Theory of Computing, Graduate Surveys* 8, pp. 1–55. DOI: 10.4086/toc.gs.2017.008. URL: https://doi.org/10.4086/toc.gs.2017.008.

Nisan, N. and A. Wigderson (1995). "Lower bounds on arithmetic circuits via partial derivatives". In: *Proceedings of IEEE 36th Annual Foundations of Computer Science*, pp. 16–25.

Pandey, Anurag, Nitin Saxena, and Amit Sinhababu (2018). "Algebraic independence over positive characteristic: New criterion and applications to locally low-algebraic-rank circuits". In: *Comput. Complex.* 27.4, pp. 617–670. DOI: 10.1007/s00037-018-0167-5. URL: https://doi.org/10.1007/s00037-018-0167-5.

Raz, Ran (Apr. 2009). "Multi-Linear Formulas for Permanent and Determinant Are of Super-Polynomial Size". In: *J. ACM* 56.2. ISSN: 0004-5411. DOI: 10.1145/1502793.1502797. URL: https://doi.org/10.1145/1502793.1502797.

Saptharishi, Ramprasad (2020). "A survey of known lower bounds in arithmetic circuit complexity". In: URL: https://github.com/dasarpmar/lowerbounds-survey.

Shpilka, Amir and Amir Yehudayoff (2010). "Arithmetic Circuits: A survey of recent results and open questions". In: *Foundations and Trends in Theoretical Computer Science* 5.3-4, pp. 207–388. DOI: 10.1561/0400000039. URL: https://doi.org/10.1561/0400000039.

Strassen, Volker (1973). "Vermeidung von Divisionen." ger. In: *Journal für die reine und angewandte Mathematik* 264, pp. 184–202. URL: http://eudml.org/doc/151394.

— (1974). "Polynomials with Rational Coefficients Which are Hard to Compute". In: *SIAM J. Comput.* 3, pp. 128–149.

Tavenas, Sébastien (2015). "Improved bounds for reduction to depth 4 and depth 3". In: *Inf. Comput.* 240, pp. 2–11. DOI: 10.1016/j.ic.2014.09.004. URL: https://doi.org/10.1016/j.ic.2014.09.004.

Valiant, L. G. (1979). "Completeness Classes in Algebra". In: *Proceedings of the Eleventh Annual ACM Symposium on Theory of Computing*. STOC '79. Atlanta, Georgia, USA: Association for Computing Machinery, 249–261. ISBN: 9781450374385. DOI: 10.1145/800135.804419. URL: https://doi.org/10.1145/800135.804419.

Valiant, Leslie G. (1982). "Reducibility by algebraic projections". In: *Logic and Algorithmic : International Symposium in honour of Ernst Specker*. Vol. 30, pp. 365–380.

Valiant, Leslie G. et al. (1983). "Fast Parallel Computation of Polynomials Using Few Processors". In: *SIAM J. Comput.* 12.4, pp. 641–644. DOI: 10.1137/0212043. URL: https://doi.org/10.1137/0212043.

Wagner, Klaus W. (1986). "The Complexity of Combinatorial Problems with Succinct Input Representation". In: *Acta Inf.* 23.3, pp. 325–356. DOI: 10.1007/BF00289117. URL: https://doi.org/10.1007/BF00289117.