Polynomial Identity Testing of Non-Commutative Circuits

Comprehensive Report

Submitted in partial fulfillment of the requirements of
BITS F422T Thesis

By

Anagha G
ID No. 2020B4A70928H

Under the supervision of:

Prof. Nitin Saxena
&
Prof. Pratyusha Chattopadhyay

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCES, PILANI,
HYDERABAD CAMPUS
April 2025

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE PILANI, HYDERABAD CAMPUS

Abstract

MSc. Mathematics (Hons.)

Polynomial Identity Testing of Non-Commutative Circuits

by Anagha G

The algebraic model of computation has gained a lot of attention in the past few decades, both due to simplicity and connections to open problems in boolean circuit complexity.

The problem of *Polynomial Identity testing* is very important due to far reaching connections with problems like primality testing, multivariate factorization. Identity testing is the problem of testing whether the given arithmetic circuit computes the zero polynomial identically. It is also interesting to study because so far we only have a BPP algorithm for PIT. Derandomizing PIT is a long standing open question, and has strong implications for lower bounds.

The motivation of this thesis is to obtain a randomized Polynomial Identity Testing Algorithm for some restricted classes of non-commutative arithmetic circuits with exponential degree. Although we have an efficient randomized algorithm for non-commutative arithmetic circuits with bounded degree, obtaining an algorithm for the exponential degree case remains a long standing open question.

We consider the Non-Commutative PIT problem in the black-box setting, wherein we are not allowed access to the internal structure of the circuit, but can only evaluate the circuit at different points. We direct our focus towards the non-commutative Algebraic Branching Programs (ABPs)

The study of sparse black-box PIT by Arvind et al, and rank concentration ideas first introduced by Saxena et al motivate us to hypothesize rank concentration for non-commutative circuits.

Acknowledgements

I am very grateful to my advisor Prof. Nitin Saxena for mentoring me over the past year and clearing all my doubts. His encouragement and insights have helped me develop a better understanding of the problem at hand. I would also like to thank him for supporting me to attend numerous workshops during my stay here. Most of the work in this thesis is joint work with Foram Lakhani. I would like to thank her for her continued assistance throughout the year, and for being available for discussions and clearing my mostly stupid doubts.

I would also like to thank my supervisors back at BITS Hyderabad, Prof. Venkatakrishnan Ramaswamy and Prof. Pratyusha Chattopadhyay. Prof. Pratyusha's engaging classes helped me consider algebra as a career and as a motivating principle for life, and I am truly grateful to her for that.

I would also like to thank IIT Kanpur for the excellent facilities provided to me over the past semester. I am particularly thankful to Rajesh ji from the office for sorting out all my administrative hassles, be it in the department or pertaining to accommodation.

And last, but not the least, I would like to thank all my friends who have stood by me and supported me throughout my college life. In particular, I would like to thank Manav, Shriya, Akshat, Atharva and Srivastav for being reliable fair weathered friends to whom I could turn to no matter what. I would also like to thank my family for giving me such a privileged life in which I could consider mathematics as a career.

Contents

List of Figures

Abbreviations

PIT	**P**olynomial **I**dentity **T**esting
ABP	**A**lgebraic **B**ranching **P**rogram

Notation

$\mathbb{Z}$	Ring of Integers
$\subseteq$	Inclusion
$\setminus$	Difference in the set theoretic sense
$\mathbb{M}_n(R)$	$n \times n$ matrix algebra over the ring R
$\times$	Cartesian Product
$R[x]$	Polynomial ring
$R[[x]]$	Power series ring
$\mathbb{F}\langle X \rangle$	Free Algebra over the field $\mathbb{F}$
$\mathbb{H}_k$	Hadamard Algebra of dimension k
$Z(R)$	Center of a ring R

Chapter 1

Introduction

1.1 Introduction

Computers are now integral to nearly every aspect of modern life, performing a vast array of tasks across science, industry, and everyday decision-making. At the heart of this computational power lies the design and analysis of algorithms: structured, rule-based procedures that solve problems efficiently.

Yet, computers are constrained by time and memory, and not all problems are equally easy to solve. Computational Complexity Theory abstracts away from implementation details to study the intrinsic difficulty of problems, classifying them based on the resources—such as time, space, or randomness required to solve them.

To provide a universal and rigorous foundation, complexity theory typically uses the Turing machine as its canonical model of computation. This model allows us to define well-known complexity classes such as:

- **P**: problems solvable in deterministic polynomial time,
- **NP**: problems for which solutions can be verified in polynomial time.

The central open question of the field, "Does P = NP?", encapsulates the mystery of whether every efficiently verifiable problem can also be efficiently solvable.

As the field has matured, new resources such as non-determinism, randomness, interaction, and communication have been incorporated into the theory, leading to a broader and richer landscape of computational models.

Beyond Turing machines, other computational models, such as probabilistic Turing machines and Boolean circuits, classify problems based on their specific resources.

Among these, a particularly elegant and powerful framework emerges when we consider algebraic computation, where the inputs and outputs are polynomials rather than binary strings. In this setting, the natural computational model is the arithmetic circuit, which captures how polynomials can be built using operations like addition and multiplication. Algebraic models offer a rich interplay between algebra, geometry, and computation. They allow us to ask structural questions about polynomials, leading to a deeper understanding of computational hardness.

1.2 Algebraic Complexity Theory

In this thesis, we will study problems of an algebraic nature. Informally, an algebraic problem is one where the inputs are elements of a field (such as $\mathbb{Q}, \mathbb{R}$, or $\mathbb{F}_p$), and the task is to compute or decide some algebraic property of the input,often involving polynomials.

Polynomials play a central role in algebraic complexity theory. Many natural questions about polynomials, such as factorization, equivalence, or identity testing, are both mathematically rich and computationally meaningful. This thesis will focus on one of the most fundamental of these: Polynomial Identity Testing (PIT), the problem of checking whether a given polynomial is identically zero.

Algebraic Complexity Theory is a subfield of computational complexity theory that studies the complexity of algebraic problems, especially those involving multivariate polynomials. It seeks to understand how efficiently polynomials can be computed and how structural properties of polynomials relate to computational resources.

In this context, the most natural model of computation is the arithmetic circuit. Just as Boolean circuits compute functions over $\{0,1\}^n$, arithmetic circuits compute polynomials over a field $\mathbb{F}$. An arithmetic circuit is a directed acyclic graph where internal nodes

(gates) compute either addition or multiplication, and the leaves are labeled by variables or field constants.

Arithmetic circuits allow us to study complexity questions such as:

- How large (or deep) does a circuit need to be to compute a given polynomial?
- Are there families of polynomials that require super-polynomial size to compute?
- Can we efficiently determine whether a given circuit computes the zero polynomial?

These questions form the heart of algebraic complexity theory and are deeply connected to some of the biggest open problems in theoretical computer science, such as the algebraic analog of $\mathbf{P} \neq \mathbf{NP}$ (the VP vs VNP conjecture).

This thesis will focus particularly on the PIT problem, and explore its connections to restricted models of arithmetic circuits- especially in the non-commutative setting.

1.3 Polynomial Identity Testing

Among the various problems studied in algebraic complexity, one particularly central and well-studied question is Polynomial Identity Testing (PIT)- the task of determining whether a given arithmetic circuit computes the zero polynomial. PIT has deep connections to circuit lower bounds, derandomization, and algebraic proof systems. While the general case remains open, many restricted models admit efficient (sometimes even deterministic) PIT algorithms. In this thesis, we will explore PIT in the context of non-commutative and structured arithmetic circuits, with a focus on understanding both their computational power and the complexity of testing identities within them.

1.4 Organization of the Thesis

In Chapter 2, we cover the necessary preliminaries and basic definitions. Chapter 3 introduces the reader to basic concepts in non-commutative algebra and the theory

of Polynomial Identity Algebras. Chapter 4 presents a structural result about non-commutative rational functions. Chapter 5 introduces the problem of Polynomial Identity Testing (PIT). Chapter 6 surveys recent advances in PIT in the non-commutative setting. Chapter 7 discusses the Rank Concentration framework introduced by Agrawal, Saha, and Saxena, which serves as an inspiration for our work. Chapter 8 describes our attempt to adapt the rank concentration idea for the non-commutative setting. Finally, Chapter 9 presents the conclusion and some open problems.

Chapter 2

Preliminaries

In this chapter, we introduce some notation that will be used in further chapters, as well as preliminary material required to digest the contents of the thesis. We also direct the reader to additional material wherever necessary.

2.1 Notation

- We use $\mathbb{F}[x_1, x_2, \cdots x_n]$ to denote the ring of polynomials over field $\mathbb{F}$.
- We use $\mathbb{F}\langle x_1, x_2, \cdots, x_n\rangle$ to denote the free algebra of non-commuting polynomials. Here, the monomials can be interpreted as *words* over $\mathbb{F}\langle x_1, x_2, \cdots, x_n\rangle$.
- Let m be a monomial and f be a polynomial. The coefficient of m in f is denoted by $\text{coeff}_m(f)$
- The set of all monomials in f such that $\text{coeff}_m(f) \neq 0$ is called the *support* of f.
- The cardinality of *support* of f is called its *sparsity*. In other words, it is the number of non-zero monomials in f.

2.2 Model of Computation

We fix an underlying field $\mathbb{F}$.

Arithmetic circuits are a standard model for computing polynomials. More formally, an arithmetic circuit $\mathcal{C}$ over $\mathbb{F}$ and the set of variables $\bar{x} = \{x_1, x_2. \cdots x_n\}$ is a directed acyclic graph with a unique sink- the **output gate**. The nodes of the graph are called gates. The in-degree of a gate is called fan-in and the out-degree its fan-out.

Each of the source vertices (the *input gates* which have fan-in 0 are labelled by either variables or field constants. All the other *internal* nodes are labelled either by $+$' or $'\times'$ and perform addition or multiplication over $\mathbb{F}$. The edges of the graph are called wires, which may carry weights from $\mathbb{F}$. Wires without labels are assumed to have weight 1.

Each gate can be recursively interpreted as computing a polynomial:

- Input gates compute their own label (a variable or constant).
- A $+$ or $\times$ gate computes the sum or product (respectively) of the polynomials computed by its children, multiplied by the constants on the incoming wires.

We assume without loss of generality that the circuit is layered- that is, wires only connect gates between successive layers. Additionally, we assume the circuit alternates between $+$ and $\times$ layers. Note that one can always restructure the circuit to have this form..

An arithmetic circuit is called a formula if every internal gate has fan-out one. In other words, the DAG is a tree. A **homogeneous** arithmetic circuit is one in which every gate computes a homogeneous polynomial.

$$f(x, y, z) = (x + y) \times z$$

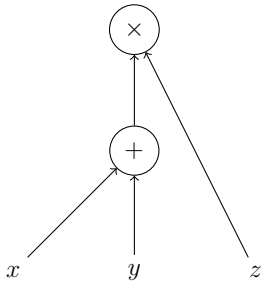

FIGURE 2.1: A simple arithmetic circuit for the polynomial $P(x) = (x + y) \times z$.

Note that the arithmetic circuit computing $P(x)$ is also an arithmetic formula.

The following are some parameters associated with arithmetic circuits:

1. The **size** of an arithmetic circuit is the number of nodes (and edges) in the circuit
2. The **depth** of an arithmetic circuit is the length of the longest directed path from the root (the output gate) to any leaf node.

The **syntactic** degree of the polynomial is the highest degree of any monomial computed by the circuit at any gate. The **degree** of the circuit is the syntactic degree of the polynomial computed at the output gate. We can compute this by computing the degree at each node in a recursive fashion. Note, that the syntactic degree might not be the same as the actual degree of the polynomial due to cancellations. For instance, if $f = xy - yx$, then the syntactic degree is 2, whereas the degree of the computed polynomial is actually 0.

Thus, arithmetic circuits are a concise way of representing polynomials. A polynomial that is computed by a size s arithmetic circuit can have degree $2^{O(s)}$. For example, the circuit computing x^{2^s} has a size $O(s)$ circuit by virtue of repeated squaring.

2.3 Non-Commutative Arithmetic Circuits

We are interested in a variant of the standard arithmetic circuit model: *non-commutative* arithmetic circuits. The study of non-commutative computation was introduced by Hyafil [Hya77] and later developed in more detail by Nisan [Nis91].

These circuits compute polynomials over the free non-commutative algebra $\mathbb{F}\langle x_1, x_2, \cdots, x_n\rangle$, where the variables do not commute-in general, $xy \neq yx$.

Restrictions on arithmetic circuits can arise in different ways. One natural way is via the interpretation of the circuit: in the non-commutative setting, we assume the variables live in a non-commutative world. This is implemented by assigning a left and right label to the children of each multiplication gate and computing accordingly.

If the circuit has size s, the degree of the computed polynomial can be as large as 2^s, and the number of distinct monomials can be up to 2^{2^s}-highlighting how expressive such circuits can be, even with small size.

Such circuits allow for fewer algebraic cancellations, making the computation inherently harder. This increased rigidity turns out to be a strength: non-commutative models are better understood from a lower bounds perspective, and have led to several strong results that remain out of reach in the commutative setting.

2.4 Algebraic Branching Programs

While trying to prove lower bounds for non-commutative formulas, Nisan [Nis91]introduced the Arithmetic Branching Program (ABP) model.

Definition 2.1. An ABP over a field $\mathbb{F}$ is a directed acyclic layered graph along with a vertex set V and an edge set $E = E_1 \sqcup E_2 \sqcup \cdots \sqcup E_d$, where $E_i \subseteq V_{i-1} \times V_i$, along with a set of labels $L_1, L2, \ldots, L_d$ such that each $L_i : E_i \to \mathbb{F}[x]$. Each label is a linear or constant polynomial in $\mathbb{F}[x]$.

We define the label as a function

$$L : E \to \mathbb{F}[x]$$
$$L|_{E_i} = L_i$$

- The vertices are thus partitioned into $q+1$ layers, including the source and the sink.
- Each edge e goes from layer V_{i-1} to a layer V_i for some $i \in [q]$, and is labelled with an element from L_i

The polynomial computed by an ABP is now of the form $\sum_{\gamma \in path(s,t)} \prod_{e \in p} L(e)$

We can define the following parameters for an ABP:

- width (w) = $\max_i |V_i|$
- The size of the ABP is the number of vertices = $w^2 q$.

Given above is a simple 2 layered ABP computing the polynomial $x_1x_3x_7 + x_1x_5x_8 + x_2x_4x_7 + x_2x_6x_8$.

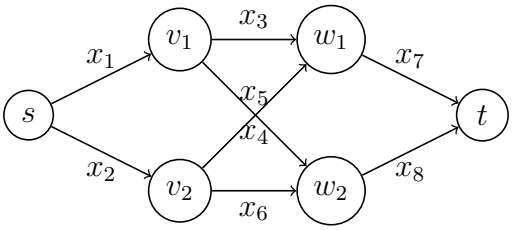

FIGURE 2.2: A 2-layered Algebraic Branching Program (ABP)

IMM Representation

ABPs are equivalent to the model of Iterated Matrix Multiplications (IMMs). This follows from the fact that the sum over all paths in a graph can be represented by an iterated matrix multiplication. Suppose the set of nodes in some layer V_i is $\{v_{i,j} : j \in [w]\}$. Then the polynomial computed by the ABP is the same as the polynomial computed by $c^\intercal Fd$, where $c, d \in \mathbb{F}^{w \times 1}$ and $F = \prod_{i=1}^{q} F_i$, where each F_i is a $w \times w$ matrix for $1 \leq i \leq q$.

Since then, various specializations of the ABP model have emerged, such as the Read-Once Algebraic Branching Program (ROABPs)- which are ABPs wherein the edge weights in different layers are univariate polynomials in *distinct* variables, and commutative ROABPs.

2.5 Set Multilinear Circuits

Set multilinear circuits are a subclass of ROABPs. More precisely, they are circuits of the form

$$C(x) = \sum_{i=1}^{k} \prod_{j=1}^{q} l_{ij}(x_j)$$

where $x_1, x_2, \ldots, x_q$ are disjoint sets of variables and each $l_{ij}(x_j)$ is a linear polynomial in x_j for each i and j.

Hadamard Product representation

A polynomial $f(x)$, which can be represented by a set multilinear circuit can be seen as a polynomial over the Hadamard algebra using a suitable dot product. This property

was heavily used by [ASS13] to develop a deterministic black-box PIT for set-multilinear circuits. We will see a short overview of this in 6.

Chapter 3

Non-Commutative Algebra: A Primer

3.1 Some Basic (Non-Commutative) Algebra

Wedderburn-Artin Theory forms a very central part of non-commutative ring theory.In this chapter we introduce some important definitions and results in non-commutative algebra which will mostly be used in Chapter 4. Throughout this chapter and the next, we mostly follow TY Lam's book [Lam01], Drensky and Formanek's notes [DF04], Artin's book [Art11] and notes [Art99] . Let K be an infinite field.

Notes

- A ring refers to a not-necessarily commutative ring with unity. We denote a ring by the symbol R usually.
- A subring of a ring R is a subset of R that in itself is a ring with unity.
- An ideal I of R refers to a *two-sided* ideal of R.
- Two sided ideals- quotient rings- subjective homomorphism
- Let A be a commutative ring. An A-algebra is a ring R along with a homomorphism from A to $Z(R)$.

- Let R be a non-commutative ring. Then $R\langle x_1, \ldots, x_n \rangle$ is called a *free associative algebra* in R over n variables.

Simple Rings

Let $R(\neq 0)$ be a ring. R is a simple ring if it has no non-trivial two sided ideals; that is, it is doesn't have any two-sided ideals apart from 0 and R. This means that $\forall r(\neq 0) \in R$, the ideal generated by r is R. However, one can observe that a simple ring may have have no non-trivial right ideals but may have non-trivial left ideals, for example, the matrix ring. One can observe the following:

1. $R(\neq 0)$ is a simple ring if and only if there exists an equation $\sum b_i a c_i = 1$ for appropriate $b_i, c_i \in R$.
2. The center of a simple ring is a field.
3. (Corollary) R is simple if and only if R is a field, in case R is commutative.

Definition 3.1 (Dimension)**.** Let R be a simple ring with the center F. Then $\dim_F R = D$ is simply the vector space dimension of of R over F.

Prime Rings

The notion of *primeness* can be extended to the non-commutative setting.

Definition 3.2 (Prime Ring)**.** Let R be a non-commutative ring. Let $a, b \in R$ such that $aRb = \{0\}$. Then R is prime if and only if $a = 0$ or $b = 0$.

A non-commutative ring is said to be prime if and only if the zero ideal is a prime ideal. Equivalently,

1. All non-zero right ideals are faithful as right R-modules
2. All non-zero left ideals are faithful as left R-modules

Recall that a module M over a ring R is said to be *faithful* if $\forall a, b \in R$, $\exists m \in M$ such that $am \neq bm$.

We will make use of the following facts:

1. Any domain is a prime ring
2. Any simple ring is a prime ring
3. Any matrix ring over an integral domain is prime

Algebras

We can make some quick analogous definitions for algebras. Let k be a field.

- A k-algebra is an associative $k-$algebra with unity. It is not necessarily commutative.
- A k-algebra is called simple if it has no proper two-sided ideals apart from (0).
- A k-algebra A is said to be central if $Z(A) = k$.
- If A is simple as well, then it called a central simple algebra.
- A k-algebra D is called a division algebra if every non-zero element has a multiplicative inverse.

3.2 Polynomial Identity Algebras

In this section, we present the basic ideas in the theory of Polynomial Identity Rings (PI-Rings). As earlier, let R be a not-necessarily commutative ring with unity 1. Further, we assume all ring homomorphism are unitary: units map to units.

Definition 3.3 (Polynomial Identity)**.** Let A be a commutative ring, and let R be an A-algebra. Let $f(x_1, x_2, \cdots, x_n) \in A\langle X \rangle$. Then f is said to be a *polynomial identity* for R if $f(r_1, r_2, \cdots, r_n) = 0 \; \forall r_1, r_2, \cdots, r_n \in R$

If the algebra R satisfies a non-trivial polynomial identity $f = 0$, then R is called a *PI-Algebra*

Some Examples

1. Every commutative ring satisfies $[x, y] = xy - yx$.
2. The free algebra $K\langle X\rangle$ does not satisfy any polynomial identity
3. Any boolean algebra satisfies $x^2 - x$
4. $M_2(K)$ satisfies the **Hall Identity**: $[[x, y]^2, z] = (xy - yx)^2 z - z(xy - yx)^2$

3.2.1 T-Ideals

Definition 3.4 (T-Ideal)**.** The set $T(R)$ of all polynomial identities of R is a two sided ideal of $A\langle X\rangle$ and is called the T-Ideal of R.

One can observe that T-ideals are invariant under substitutions from $A\langle X\rangle$. They are also thus closed under A-endomorphisms, since any substitution can be seen as an A-algebra endomorphism.

Definition 3.5 (Primitive Ring)**.** A ring R is said to be *(left) primitive* if it has a simple faithful left R-module

Analogously one can define right primitive rings.

Some examples and properties

1. Every simple ring is both left and right primitive
2. A commutative ring is primitive if and only if it is a field
3. A division ring is always a primitive ring
4. $K\langle X\rangle$ is a primitive ring.
5. $K[X]$ is not a primitive ring because it is not a field, from (2)

Definition 3.6 (Proper polynomial)**.** A polynomial f is said to be *proper* if some coefficient in the highest degree homogeneous component of f is equal to 1.

Definition 3.7 (PI-algebras and PI-rings)**.** Let A be a commutative ring and let R be an A-algebra. Suppose there is a proper polynomial $f \in A\langle X\rangle$ such that f is a polynomial identity for R. Then,

1. R is said to be a Polynomial Identity Algebra (PI-algebra) over A.
2. If $A = \mathbb{Z}$, then R is said to be a Polynomial Identity Ring (PI-ring).

We state the following theorem from combinatorial PI theory [Ami71].

Theorem 3.8. *A PI-ring always satisfies a multilinear proper PI*

Suppose a proper polynomial f is a polynomial identity for some A-algebra R. Then, there exists a proper multilinear polynomial g, with the same degree of f, which is also an identity for R.

In other words, a PI-ring always satisfies some multilinear proper polynomial identity. We use this to prove the following corollary, which comes of use later.

Corollary 3.9. *Suppose R is a PI-ring, then so is $R[T]$, where T is a set of central indeterminates over R*

Proof. By Theorem 3.8, R satisfies some multilinear proper polynomial $f(x_1, x_2, \cdots, x_n) \in \mathbb{Z}\langle X\rangle$. Since f is multilinear, any evaluation of f on $R[T]$ can be written as a sum of evaluations where each variable is replaced by a product of the form $r_i\mu_i$, where $r_i \in R$ and μ_i is a monomial in T. Due to multilinearity, this decomposes in a clean fashion as: $f(r_1\mu_1, \ldots, r_n\mu_n) = f(r_1, \ldots, r_n) \cdot \mu_1 \cdots \mu_n$. But R satisfies the identity f, and thus $f(r_1, \cdots, r_n) = 0$ and so $f(r_1\mu_1, \ldots, r_n\mu_n) = 0$.

And hence f is also a PI for $R[T]$.

□

3.3 Amitsur-Levitzki Theorem

Definition 3.10 (Standard Polynomial)**.** The Standard Polynomial of degree n is

$$s_n(x_1, \cdots, x_n) = \sum\{sgn(\sigma)x_{\sigma(1)} \cdots x_{\sigma(n)} : \sigma \in S_n\}$$

The standard polynomial is a multilinear and alternating polynomial (over all its variables).

Idempotents

The proof presented below uses some new terminology that we introduce here. Let R be a ring.

1. An idempotent of R is an element $r \in R$ such that $r^2 = r$.
2. Let a and b be two idemptotent elements. They are said to be *orthogonal* if $ab = 0 = ba$.
3. Let $a \in R$ be an idempotent. a is called *centrally idempotent* if $\forall r \in R, ar = ra$. Equivalently, $a \in Z(R)$.
4. A *primitive idempotent* of R is then a nonzero idempotent $a \in R$ such that aR cannot be written as a direct sum of two non-zero submodules. Equivalent, $\nexists e, f$, which are nonzero orthogonal idempotents such that $a = e + f$.
5. 0 and 1 are trivially idempotent

We also need a few properties of the standard polynomial.

Lemma 3.11. *The following are some properties of the standard polynomial s_n. Use x_i' to indicate that x_i variable is deleted from the input.*

1. $s_{n+1}(x_1, \cdots, x_{n+1}) = \sum \pm x_1 s_n(x_1, \cdots, x_i', \cdots, x_{n+1})$
2. $s_n(\dots, x_i, \dots, x_j, \dots) = -s_n(\dots, x_j, \dots, x_i, \dots)$
3. $s_{2n}(1, x_2, \dots, x_{2n}) = 0$
4. $s_{n+1}(x_1, x_2, \dots, x_{n+1}) = \sum \pm x_1 s_n(x_1, \dots, x_i', \cdots x_{n+1})$

Proof. 1. We essentially want to show that if an algebra A satisfies s_n, it also satisfies s_{n+1}. Observe that $\sum \pm x_1 s_n(x_1, \cdots, x_i, \cdots, x_{n+1})$ is equal to the sum of those terms in $s_{n+1}(x_1, \cdots, x_{n+1})$ that have x_i on their left.

2. We want to show that swapping two variables in s_n only changes the sign. Consider the transposition $\tau = (I, j) \in S_n$ that swaps i and j. Observe that swapping two variables is equivalent to composing each permutation σ with the transposition τ applied to the inputs. Hence

$$s_n(\dots, x_j, \dots, x_i, \dots) = \textstyle\sum_{\sigma \in S_n} \operatorname{sgn}(\sigma) x_{\tau(\sigma(1))} \cdots x_{\tau(\sigma(n))}$$

See that $\sigma \mapsto \tau \circ \sigma$ is a bijection of S_n onto itself. Change variables due to invertibility: $\operatorname{sgn}(\sigma) = \operatorname{sgn}(\tau \circ \sigma') = \operatorname{sgn}(\tau)\operatorname{sgn}(\sigma')$.

By substitution, $x_{\tau(\sigma(1))} \cdots x_{\tau(\sigma(n))} = x_{\sigma'(1)} \cdots x_{\sigma'(n)}$

Thus,

$$\begin{aligned} s_n(x_1, \dots, x_j, \dots, x_i, \dots) &= \sum_{\sigma' \in S_n} \operatorname{sgn}(\tau) \operatorname{sgn}(\sigma') \, x_{\sigma'(1)} \cdots x_{\sigma'(n)} \\ &= \operatorname{sgn}(\tau) \sum_{\sigma' \in S_n} x_{\sigma'(1)} \cdots x_{\sigma'(n)} \\ &= \operatorname{sgn}(\tau) s_n(x_1, \dots, x_n) \end{aligned} \tag{3.1}$$

As τ is a transposition, $\operatorname{sgn}(\tau) = -1$ and the assertion holds.

3. Note that for each ordering of $x_2, \dots, x_{2n}$, 1 can be placed in $2n$ positions, of which half evaluate to positive and half to negative. So everything cancels out.

4. The RHS $\sum \pm x_1 s_n(x_1, \dots, x_i', \cdots x_{n+1})$ is equal to the sum of those terms in the LHS that start with x_1 (to the left). Hence the assertion follows.

□

Lemma 3.12. *1. Let s''' be the sum of the terms in $s_n(x_1, \dots, x_n)$ in which the product $x_1x_2x_3$ occurs. Then $s''' = s_{n-2}(x_1x_2x_3, x_4, \cdots, x_n)$*

2. Use x_i' to indicate that x_i variable is deleted from the input. Let s'' be the sum of the terms in $s_n(x_1, \dots, x_n)$ in which the product x_1x_2 occurs. Then $s'' = \pm s_{n-2}(x_3, x_4, \cdots, x_n)x_1x_2 + \sum_3^n (\pm) s_{n-2}(x_1x_2x_i, x_3, \dots, x_i', \dots, x_n)$

Proof. 1. We need to confirm whether the terms in s''' and $(x_1x_2x_3, x_4, \cdots, x_n)$ that correspond to each other have the same sign. To see this, we study the number of transpositions. In s''', $x_1x_2x_3$ can be brought to the leftmost positions be shifting through the, say r preceding elements. The number of transpositions needed is $3r$. In $s_{n-2}(x_1x_2x_3, x_4, \cdots, x_n)$, you need only r transpositions. Then, $(-1)^{3r} = (-1)^r$.

2. $\pm s_{n-2}(x_1x_2x_3, x_4, \cdots, x_n)$ has all those terms in s'' with x_1x_2 followed by x_i. This follows from the variable flip rule 2 and from the fact proved above. Then, $\pm s_{n-2}(x_3, x_4, \cdots, x_n)x_1x_2$ has all those terms with x_1x_2 followed by nothing. We are done.

□

Lemma 3.13. *Suppose $n > 1$, and $M_{n-1}(K)$ satisfies s_{2n-2}. Let $\{e_{a_ib_i}\}$ be a collection of $2n$ elements of the standard basis of $M_n(K)$. Further, suppose the evaluation of s_2n on these basis elements is non-zero. Define the following quantity:*

For each $u = 1, 2, \ldots, n$, let $\nu(u)$ be the number of times u occurs in the subscript in $\{e_{a_ib_i}\}$. Then $\nu(u)$ has only the following three possible configurations:

- $3, 5, 4, 4, 4, \ldots$
- $3, 3, 6, 4, 4, \ldots$
- $4, 4, 4, 4, 4, \ldots$

Proof. **Claim:** $\forall u, \nu(u) \geq 3$

We show this by eliminating the other cases.

- Suppose $\nu(u) = 0$ for some u. See that $\{e_{a_ib_i}\} \subseteq M_n(K)^{(u)} \cong M_{n-1}^{(u)}$. By assumption, $M_{n-1}(K)$ satisfies s_{2n-2}. It also satisfies s_{2n} by 4. But since s_{2n} is not an identity for our chosen set, this cannot happen.
- Suppose $\exists u$ such that $\nu(u) = 1$. Further suppose e_{iu} is in our chosen set. See that only the monomials that survive in $s_{2n}(e_{a_1b_1}, \ldots, e_{a_{2n},b_{2n}})$ must have e_{iu} on the right. Thus, $s_{2n}(e_{a_1b_1}, \ldots, e_{a_{2n},b_{2n}}) = s_{2n-1}(e_{a_1b_1}, \ldots, e'_{iu}, \ldots, e_{a_{2n},b_{2n}})e_{iu}$. Now all the values in $s_{2n-1}(\ldots)$ are contained in $M_n(K)^{(u)}$, and we reach a similar contradiction. Mimic a parallel argument by assuming e_{ui} is one of our chosen elements.
- Suppose $\exists u$ such that $\nu(u) = 2$. Observe that $s_{2n}(\ldots) = 0$ if the subscript u occurs in any of the following three ways:
  - e_{uu}

- e_{iu} and e_{ju}
- e_{ui} and e_{uj}

for any i and j. This leaves us with only one combination: e_{ui} and e_{ju}. The non-zero monomials in $s_{2n}(\dots)$ then occur in one of the two possible ways.

- Suppose e_{uj} is on the leftmost end and e_{iu} is on the rightmost end. The sum of such terms is 0 as s_{2n-2} is an identity for $M_{n-1}(K)$.
- If we have e_{iu} followed immediately by e_{uj}, this breaks down to e_{ij} and these terms also go to 0 by 1. Thus we are done.
- Since $s_{2n}(\dots) \neq 0$, some monomial survives. Due to symmetry, either all $\nu(u)$ are even or all but two are even. We want to satisfy $\sum_1^n \nu(u) = 4n$, and there are exactly $4n$ subscripts in our chosen set. This completes the proof.

□

While it seems rather irrelevant, the above combinatorial observation is what ties together the following proof of the Amitsur Levitzki theorem!

Theorem 3.14 (Amitsur-Levitzki Theorem)**.** *Let K be a field. Then,*

1. *$s_{2n}(x_1, \cdots, x_{2n})$ is a polynomial identity for $M_n(K)$*
2. *$M_n(K)$ does not satisfy any polynomial identity of degree $\leq 2n-1$*

Proof. 1. We give an inductive proof that slightly varies from the original, based on [Pas11]. It is largely combinatorial.

The proof is by induction on n. For $n = 1$, $M_1(K)$ is commutative, and $s_2(x, y) = xy - yx$.

For the inductive step, suppose $n > 1$ and $M_{n-1}(K)$ satisfies s_{2n-2}.

Step 1: We show that $s_{2n}(e, f, E, \cdots, E) = 0$ for orthogonal primitive idempotents $e, f \in M_n(K)$.

We use a standard basis $\{e_{ij}\}$ with $e = e_{11}$ and $f = e_{22}$. Suppose $s_{2n}(e, f, E, \cdots, E) \neq 0$. By a linearity argument, some term $s_{2n}(e_{a_1b_1}, \dots, e_{a_{2n}b_{2n}}) \neq 0$ where $e_{ij} \in$

$\{e_{11}, e_{22}, \dots\}$. By Lemma 3.13, at most one index has $\nu(u) > 4$, where $\nu(u)$ is the number of occurrences of row or column index u. Hence at least one of $\nu(1) \leq 4$ or $\nu(2) \leq 4$. Now replace e_{11} by $1 - \sum_2^n e_{ii}$. Invoking 3, you can see that $s_{2n}(1, \dots,) = 0$. Then, in the remaining terms we get $\nu(u) < 2$. But again by Lemma 3.13, they must vanish, leading to a contradiction.

Step 2: Show that $s_{2n}(e, E, E, \cdots, E) = 0$. where $e \in M_n(K)$ is a primitive idempotent element.

Take $e = e_{11}$ as an element of the standard basis. We now evaluate $s_{2n}(x_1, \dots, x_{2n})$ where $x_1 = e_{11}$, and the rest are arbitrary basis elements. Consider a term in the polynomial: $x_{\sigma(1)}x_{\sigma(2)} \dots x_{\sigma(2n)}$. For this term to be nonzero, the product of the basis elements need to form a chain in the following way: if $x_1 = e_{ab}$ then the product is nonzero only if each index matches: $e_{a_1b_1}e_{a_2b_2} \dots e_{a_{2n}b_{2n}} \neq 0 \implies b_i = a_{i+1}\ \forall i$.

There are two possible cases for the basis elements:

(a) All $x_i \in \text{Span}\{e_{i1}, e_{1j}\}$, that is every basis element has row or column index as 1. But there are only $2n - 1$ basis elements left. By the pigeonhole principle, some index appears at least twice. By 2, this term is zero, and the entire term vanishes.

(b) Some $x_i = e_{ij}$ with $i, j \neq 1$, that is there is some basis element that does not involve the index 1. Then this basis element is orthogonal to e_{11} unless there is some overlap due to chaining. But due to Step 1, we saw that placing two primitive orthogonal idempotents together in the standard polynomial anniholates it. Thus replacing e_{ij} with some linear combination of such elements vanishes all the terms.

Hence the assertion holds.

Step 3: Show that $s_{2n}(E, E, E, \cdots, E) = 0$. where $e \in M_n(K)$

Show that the polynomial vanishes on the whole algebra. By linearity, it suffices to show that it is zero on the standard basis. We again have two cases: Let e_{ab} denote any basis element.

(a) Suppose e_{ab} with $a = b$. These are diagonal basis elements, and are primitive idempotents. By step 2, the standard polynomial vanishes here.

(b) Suppose e_{ab} with $a \neq b$. These are the off-diagonal entries. Write any e_{ab} as $e_{ab} = (e_{aa} + e_{bb}) - (e_{aa} + e_{bb} - e_{ab})$, where e_{aa} and e_{bb} are primitive idempotents. Again, using linearity of s_{2n} and expanding, as well as using the result from Step 2, these terms vanish! and thus s_{2n} is zero on the basis.

2. Suppose $M_d(K)$ satisfies a polynomial identity of degree $k < 2d$. It must be that it satisfies a multilinear identity of the following form:

$$f(x_1, x_2, \cdots, x_k) = \sum_{\sigma \in S_k} \alpha_\sigma x_{\sigma(1)} \cdots x_{\sigma(k)}$$

where $\alpha_\sigma \in K$.(This follows from a linearization argument). Let $\{e_{ij}\}_{i,j=1}^{k}$ be the basis of the $k \times k$ matrix algebra. We then use staircase arguments:

$$f(e_{11}, e_{12}, e_{22}, e_{23} \cdots e_m n) = \alpha_\varepsilon e_{1m}$$

. Depending on whether k is even or odd, $m = n$ or $m = n - 1$, and ε is the identity permutation. It follows that $\alpha_\epsilon = 0$.

□

3.4 Chain Conditions

Definition 3.15 (Artinian Rings)**.** A ring R is *left Artinian* if it satisfies the descending chain condition on left ideals, that is every descending chain of left ideals is stationary.

Lemma 3.16. *Let E be a field. If R is a sub-ring of $M_n(E)$ which contains an E-basis for $M_n(E)$, then R is a prime PI-ring.*

Proof. Since R is a subring of $M_n(E)$, which is a PI-ring, R is also a PI-ring. It remains to show that R is prime.

Let $\{e_{ij}\}_{1 \leq i,j \leq n}$, where $e_{ij} \in M_n(E)$ is the matrix with 1 in the (i, j)th position and zeroes everywhere else.

We now show that $R \subseteq M_n(E)$ contains some E-basis of $M_n(E)$, it must be equal to the entire ring. Suppose $\{b_1, \cdots, b_{n^2}\}$ be an E-basis of $M_n(E)$. Then any $A \in M_n(E)$ can be written as an E-linear combination as:

$$A = \sum_{i=1}^{n^2} \lambda_1 b_i \text{ for } \lambda_i \in E.$$

Observe that the scalar matrices are the only matrices that commute with the center of the matrix ring, and thus $Z(M_n(E)) = \{\lambda I_n : \lambda \in E\}$.

However, $\{\lambda I_n : \lambda \in E\} \cong E$. In particular, $E \subseteq Z(M_n(E))$.

Since $b_i \in R$, and the scalar multiples of elements in a ring are in the ring, we get $A \in R$, and thus $R = M_n(E)$. Note that we need the scalars to be from a central subring, which E satisfies.

Matrix rings over fields are simple, and simple rings are prime. Thus $M_n(E)$ is also a prime ring.

□

3.4.1 Wedderburn-Artin Theorem

The following result is due to Wedderburn and Artin:

Theorem 3.17. *Wedderburn's Theorem Suppose R is a simple left Artinian ring. Then there exists a division ring D and an integer n such that $R \cong M_n(D)$.*

For example, suppose $R = M_n(K)$. Thus R is already a matrix algebra over a field K

For a detailed simple exposition of the theorem we direct the reader to [Bre24]. The proof makes use of idempotents that we introduced earlier,

3.5 Central Polynomials

Definition 3.18 (Central Polynomial)**.** Let A be a commutative ring, and let R be an $A-$algebra with a center C. A polynomial $f \in A\langle X\rangle$ is a central polynomial for R if the following hold:

1. $\forall r_1, \cdots, r_n \in R, f(r_1, r_2, \cdots, r_n) \in C$

2. $\exists r_1, \cdots, r_n \in R$, such that $f(r_1, r_2, \cdots, r_n) \neq 0$
3. $f(x_1, \cdots, x_n)$ does not have a constant term.

Informally, a central polynomial (for the $n \times n$ matrix algebra) can be seen as a polynomial in non-commutating variables, that is not a constant polynomial, but forms a scalar matrix on evaluation at $n \times n$ matrices.

Some properties and examples

- $(xy - yx)^2$ is a central polynomial for $M_2(K)$
- If a ring R does not satisfy any PI, then R has no central polynomials.

We introduce a polynomial called Formanek polynomial, which is used in later sections. The existence of the polynomial can be shown constructively, and we direct the reader to [For72] for the explicit construction. We present the definition below.

Definition 3.19 (Formanek Polynomial)**.** $\forall n$, there exists a polynomial $F_n = f(x, y_1, \cdots, y_n)$ which satisfy the following:

1. F is homogeneous (in the usual sense) of degree $n^2 - n$ in x, and is multilinear in $y_1, \cdots, y_n$
2. Suppose K is a field. Then F_n is a central polynomial for $M_n(K)$

The following corollary can be obtained from the construction, and we simply state it:

Corollary 3.20. *Suppose $f \in \mathbb{Z}\langle X \rangle$ is a polynomial identity for $M_n(\mathbb{Z})$. Then f is a polynomial identity for $M_r(\mathbb{Z})$ for $r < n$.*

3.6 Kaplansky's Theorem

This theorem is of historical importance because the study of PI algebras originated with this paper of Kaplansky in which he showed that primitive PI algebras are finite dimensional over their centers. This can be seen as a generalization of the fact in the commutative setting that "A primitive commutative ring is a field".

Theorem 3.21 (Kaplansky's Theorem)**.** *Let R be a primitive PI-ring. Then $R \cong M_n(D)$, where D is a division algebra finite dimensional over its center K. In other words, R is a central simple algebra that is finite dimensional over its center K.*

We state the following lemma without proof.

Lemma 3.22. *Let R be a central simple algebra that is finite dimensional over its center K. Then, $\exists n \in N$ such that*

1. $\dim_K(R) = n^2$
2. *Every PI satisfied by $M_n(\mathbb{Z})$ is satisfied by R.*
3. *No polynomial of degree $\leq 2n-1$ is a PI for R*
4. *$F_n = F(x, y_1, \cdots, y_n)$ (where F_n is the Formanek polynomial as in Definition 3.19) is a central polynomial for R.*

Definition 3.23 (PI-degree)**.** A ring R has PI-degree n if (2), (3) and (4) of Lemma 3.22 hold.

3.7 Ring of Generic Matrices

Definition 3.24 (Generic Matrix)**.** Let K be an infinite field. We denote the set of (independent) commuting indeterminates by $\{x^i_{j,k} : a \leq j, k \leq d, i \in \mathbb{N}\}$. Define a matrix $X_i = [x^i_{j,k}]$ for each $i \in \mathbb{Z}_{>0}$. By a generic matrix we mean a matrix of indeterminates. $K[\{x^i_{j,k}\}]$ is the polynomial ring over $x^i_{j,k}$.

Definition 3.25 (Ring of Generic Matrices)**.** The subalgebra generated by $\{T_i : i \in \mathbb{Z}_0\}$ is $R_d \subseteq \mathbb{M}_d(K[\{t^i_{j,k}\}])$, the ring of generic matrices. We also use the notation $K\{X_1, X_2, \cdots, X_n\} = K\{X\}$.

T-ideal

The notion of T-ideal of identities has been introduced earlier. In this section, we show an important theorem that characterizes these T-ideals.

Theorem 3.26 (Isomorphism)**.** *Let K be an infinite field. Let $K\{X\}$ be the ring of $n \times n$ generic matrices over K. Consider the map $\phi : K\langle X\rangle \to K\{X\}$, induced by $x_i \to X_i$. The kernel of ϕ is given by*

$$M(n) = \{f \in K\langle X\rangle : f \text{ is a PI for } M_n(R) \text{ for any commutative } K\text{-algebra } R\}$$

$M(n)$ is equal to the T-ideal of identities of $M_n(K)$

The main goal of this section is to build up a theorem of Amitsur that will be proved in Chapter 4. Some prior knowledge of Galois theory is assumed, and we direct the reader to [Mor96].

Definition 3.27 (Twisted Laurent Polynomial Construction)**.** The construction present here is slightly different from the one given by Amitsur. Let K be a field. Let $K[x_1, x_2, \cdots, x_n]$ be the polynomial ring over K, and let $L = K(x_1, \cdots, x_n)$ be its quotient field.

.Let ϕ be the K-algebra automorphism induced by $\phi(x_i) = x_{i+1}$. (Assume the indices go modulo n).

Then define the K-algebra $R = (L, \varphi, \sigma^{\pm 1})$ as follows:

- R is free as a left L-module. Further, R has the basis $\{\sigma^i : I \in \mathbb{Z}\}$
- Multiplication is defined as : $\sigma^i\sigma^j = \sigma^{i+j}$
- Let $a \in L$. $\sigma a = \varphi(a)\sigma$.

Theorem 3.28. *Let R be defined as above. Let C be the center of R and let $S = C \setminus \{0\}$. Then,*

1. *R has no zero divisors*
2. *Suppose L^φ denotes the fixed field (elements of L unchanged by the action of φ). Then $C = L^\varphi[a^n, a^{-n}]$*
3. *R is a free module over C and has rank n^2*

4. *$S^{-1}R$ is a division ring of dimension n^2 over the center $S^{-1}C$. Then, R has PI-degree n.*

Proof. 1. We use degree arguments. Every element in R is a finite sum of the form $\sum_i a_i\sigma^i$, where each $a_i \in L$ and only finitely many a_i are nonzero. Define the degree of the term $a_i\sigma^i$ as i, and the degree of the sum to be the highest appearing degree with a nonzero coefficient. Suppose you multiply x that has leading term $a_i\sigma^i$ and y with leading term $b_j\sigma^j$ then xy has a leading term $a_i\varphi^i(b_j)\sigma^{i+j}$.

Note that $a_i, b_j \neq 0$, and $\varphi^i(b_j) \neq 0$ as φ is a field automorphism, so it preserves nonzero elements. The product of leading terms is nonzero and hence xy is also nonzero.

2. We find all elements of R that commute with everything.

   (a) The Centralizer of L in R is $L[\sigma^{\pm n}]$

   An element $\sum_i a_i\sigma^i$ centralizes L if and only if $(\sigma_i a_i\sigma^i)c = c(\sigma_i a_i\sigma^i)$ for all $c \in L$.

   This means $\sum_i a_i\varphi^i(c)\sigma^i = \sum_i ca_i\sigma^i$ for all $c \in L$.

   Comparing coefficients, $a_i\varphi^i(c) = c(a_i)$ for all i and for all $c \in L$.

   For $a_i \neq 0$, this is equivalent to $\varphi^i(c) = c$ for all $c \in L$.

   Since φ has order n, $\varphi^i(c) = c$ for all $c \in L$ if and only if $n|i$. Therefore the centralizer of L in R is $L[\sigma^{\pm n}]$

   (b) The centralizer of σ in L is $L^\varphi[\sigma^{\pm 1}]$

   An element $\sum_i a_i\sigma^i$ centralizes σ if and only if $\sigma(\sum_i a_i\sigma^i) = (\sum_i a_i\sigma^i)\sigma$

   This gives $\sum_i \varphi(a_i)(\sigma)^{i+1} = \sum_i a_i\sigma^{i+1}$.

   Comparing the coefficients, $\varphi(a_i) = a_i$ for all i. Thus $a_i \in L^\varphi$. Hence The centralizer of σ in L is $L^\varphi[\sigma^{\pm 1}]$

   Combining the two, $C = L^\varphi[\sigma^{\pm 1}] \cap L[\sigma^{\pm n}] = L^\varphi[\sigma^{\pm n}]$

3. We know that R is built from L and a. Since φ has order n, L has dimension n over L^φ. R is thus a free left L-module on $\{a^i : 0 \leq i < n\}$.

   As a free left $L^\varphi[a^n]$ module, R has a basis $\{t_j a^k : 0 \leq j, k < n\}$ where $\{t_j\}$ is a basis of L over L^φ. Thus the rank is $n \times n = n^2$.

4. By 1 and 3, we can see that R has no zero divisors, and is finite over center C. Invoking Posner's theorem (Corollary 4.7,) $S^{-1}R$ is a central simple ring over $S^{-1}C$.

□

3.8 Some Theorems on Radicals

The Jacobson Radical can be defined in the non-commutative setting similar to the commutative setting. We use the following notion consistently.

Definition 3.29 (Jacobson Radical)**.** The Jacobson Radical $\mathcal{J}(R)$ of a ring R is the intersection of the *maximal left ideals* of R.

It is interesting to observe the following hold even in the non-commutative setting.

Lemma 3.30. *Let R be a ring, and let $\mathcal{J}(R)$ be the Jacobson radical of R. Then,*

1. $\mathcal{J}(R) = \{r \in R : 1 - ar \text{ is invertible } \forall a \in R\}$
2. $\mathcal{J}(R) = \{r \in R : 1 - ra \text{ is invertible } \forall a \in R\}$

The proof mirrors that of the commutative case.

Definition 3.31 (Subdirect Product)**.** Let R be a ring. R is said to be the subdirect product of the rings $\{R_\alpha : \alpha \in \mathcal{A}\}$ if the following hold:

1. Each R_α is isomorphic to some R/M_α of R, where each M_α is a two-sided ideal of R
2. The map $R \to \prod\{R_\alpha : \alpha \in \mathcal{A}\}$ is injective.

Definition 3.32 (Semi-primitive Rings)**.** A ring R is semi-primitive if it can be written as a *subdirect product* of primitive rings

It can be seen that a ring R is semi-primitive if and only if $\mathcal{J}(R) = 0$

Definition 3.33 (Semi-prime Rings)**.** A ring R is said to be semi-prime if it is a *subdirect product* of prime rings.

Similarly, one can also define the notion of a formal power series ring.

Definition 3.34 (Formal Power Series Ring)**.** Let R be a commutative ring. The formal power series ring over R, denoted by $R[[t]]$ is the set of all power series with coefficients in R.

Before this lemma, we recall a result from the commutative world.

Lemma 3.35 (Atiyah McDonald Chapter 1 Exercise 2.1)**.** *Let R be a ring and let $R[x]$ be the polynomial ring. Let $f = a_0 + a_1x + \cdots + a_nx^n \in R[x]$. Then, f is invertible in $R[x]$ if and only if $a_1, \cdots, a_n$ are nilpotent elements.*

Lemma 3.36. *Let R be a ring. Let $r = r_0 + r_1t + \cdots + r_nt^n \in R[t]$ where $r_0, r_1, \cdots, r_n$ are commuting entries. Then, r is invertible over $R[t]$ if and only if*

1. *r_0 is invertible in R*
2. *$r_1, r_2, \cdots, r_n$ are nilpotent elements.*

Proof. Suppose r is invertible in $R[t]$. By multiplying and expanding out, we can see that r_0 is invertible in R. Suppose $\tilde{R}$ be the commutative subring of R generated by $r_0, r_1, \cdots, r_n$ and r_0^{-1}. Then r is invertible in the power series ring $\tilde{R}[[t]]$.

To show that $r^{-1} \in \tilde{R}[t]$, notice that $\tilde{R}[t] = R[t] \cap \tilde{R}[[t]]$. The rest follows from Lemma 3.35. □

Lemma 3.37. *Let R be a prime ring, and let L be a left ideal of R. Let $r(L) = \{r \in R : Lr = 0\}$. Then,*

1. *Define $M = r(L) \cap L$. M is a two-sided ideal in L.*
2. *L/M is a prime ring.*

Proof. 1. First, we show that M is a left ideal in L. Since for any $m \in M$ and $l \in L$, we need to show that $l \cdot m \in M$. Since $m \in M$, we have $m \in L$ and $m \in r(L)$. Then, $l \cdot m \in L$ since L is a left ideal of R and $l, m \in L$. Now since $m \in r(L)$ we have $Lm = 0$. In particular, $L(lm) = (Ll)m = 0$, and thus $lm \in r(L)$. Therefore $lm \in L \cap r(L) = M$

Now we show that M is a right ideal in L. For any $m \in M$ and $l \in L$, we need to show $m \cdot l \in M$. We know that $m \in L, m \in r(L)$. Since L is a left ideal of R, we have $ml \in L$ only if $ml \in R$, which holds. Now we are left to show $ml \in r(L)$, that is $L(ml) = 0$. Let $x \in L$. We have $xml = 0$ as $m]inr(L)$ and this implies $Lm = 0$. Therefore, $L(ml) = 0$ and thus $ml \in r(L)$. Thus $ml \in L \cap r(L) = M$.

2. To show that L/M is a prime ring, we need to verify that for any two sided ideals A/M and B/M of L/M, if $(A/M)(B/M) = 0$, then either $A/M = 0$ or $B/M = 0$.

   So let A/M and B/M be two sided ideals of L/M such that $(A/M)(B/M) = 0$. Then, $AB \subseteq M$. Since $M \subseteq r(L)$, we also have $AB \subseteq r(L) \implies L(AB) = 0$. Recall that R is prime. Let $a \in A$ and $b \in B$. Now $\forall l \in L$ and $r \in R$ we have

   (a) $l(arb) = (la)rb \in (LA)(RB) = LRB \subseteq LB$ (as A is an ideal of L.

   (b) But $LRB = 0$ as $L(AB) = 0$.

   Hence $ARB \subseteq r(L)$. Now if $A \nsubseteq M, \exists a \in A$ such that $a \notin r(L)$. Since $a \notin r(L)$, $\exists l \in L$ such that $la \neq 0$. Since R is prime and $ARB \subseteq r(L)$, we have $la \cdot R \cdot B = 0 \implies laRB = 0$. Now $la = 0$ or $B = 0$. Since $la \neq 0$ by choice of a, it forces $B = 0$. But this means $B \subseteq M$, and thus $B/M = 0$.

   Mimicking this argument, if $B \nsubseteq M$, can conclude that $A/M = 0$.

□

Lemma 3.38. *Suppose that R is a prime ring that is not a domain. Then $\exists r(\neq 0) \in R$ such that $r^2 = 0$.*

Proof. Since R is not a domain, there exist $a, b \in R$ such that $ab = 0$. Now consider the element $r = ba$. If $r = 0$, then we have $ab = 0$ and $ba = 0$. Then $(aRb)^2 = 0$ since $\forall x, y \in R, (axb)(ayb) = (ax)(ba)(yb) = 0$. But since R is prime, $(aRb)^2 = 0 \implies aRb = 0$, which means $a = 0$ or $b = 0$, contradicting our choice of a and b. Thus $r = ba \neq 0$ and $r^2 = (ba)(ba) = b(ab)a = 0$.

□

Recall that an ideal is said to be nil if each of its elements is nilpotent.

Theorem 3.39 (Levitzki). *Let R be a prime PI-ring, and let R be an A-algebra. Then R contains no non-zero nil ideals.*

Proof. From , R satisfies a proper multilinear polynomial, say f such that

$$f(x_1, \cdots, x_n) = x_1 x_2 \cdots x_n + \sum\{a(\sigma) x_{\sigma(1)} x_{\sigma(2)} \cdots x_{\sigma(n)} : \sigma \in S_n, \sigma \neq 1\}$$

We induct on the degree of f. For $n = 1$, $f(x_1) = x_1$ and $R = 0$, which is true.

For the inductive step, suppose the claim is true whenever a prime PI ring satisfies a proper multilinear PI of degree $n - 1$. We will then show assuming that R has a nonzero nil ideal leads to a contradiction.

Suppose R has a nonzero nil ideal N. Pick $0 \neq a \in N$ such that $a^2 = 0$ (can do this as it is nil ideal). Also let $L = Ra$, a left ideal of R. Then, $R' = L/(r(L) \cap L)$ is a prime ring by Lemma 2. The idea is to plug in elements involving a to reduce the degree. Substitute $ar_1, r_2a, \cdots r_na$ into f where $r_1, \cdots, r_n \in R$. On expanding, this splits into two parts:

1. Terms involving a^2: These vanish as $a^2 = 0$ by assumption
2. Terms where a appears exactly once: these are the terms that survive

These terms now involve a new multilinear polynomial, call it $g(x_2, \cdots, x_n)$ of degree $n - 1$. Thus$g(x_2, \cdots, x_n)$ is a proper multilinear polynomial identity for L and thus also for R'.

By the inductive hypothesis, R' has no nonzero nil ideals. Observe that

1. $L = Ra \subseteq N$, so all the elements of L are nilpotent
2. $R' = L/(r(L) \cap L)$ is nil

R' is prime and nil, and thus $R' = 0$.

We will now conclude that $L^2 = 0$. Since $R' = 0, L \subseteq r(L)$ and hence for any $l_1, l_2 \in L, l_1 l_2 = 0$ and thus $L^2 = 0$. In particular, as $L = Ra,$, $RaRa = 0$. But thus means the square of the two sided ideal generated by a is 0. But this contradicts the hypothesis that

R is a prime ring. Thus our assumption that $N \neq 0$ is wrong. Thus R has no non-zero ideals, completing the induction

□

Theorem 3.40 (Amitsur)**.** *Let R be a ring with no nil ideals. Let t be a commuting indeterminate over R. Then R is a semiprimitive ring.*

Proof. **Claim:** Suppose $\mathcal{J}(r) \neq 0$. Then R must contain a non-zero nil ideal. Suppose $\mathcal{J}(R[t]) \neq 0$. Let n be the minimal degree of a nonzero element of the radical. Now define

$$I = \{r_n \in R : \exists r \in \mathcal{J}(R[t]) \text{ with} r = \textstyle\sum_{i=0}^{n} r_i t^i\}$$

Then one can see I is a two sided ideal of R. We now use an inductive argument to show that $r_0, r_1, \ldots, r_n$ all commute with each other. Since n is the minimal degree of any nonzero element in $\mathcal{J}(R[t]), r_n \cdot r - r \cdot r_n = 0 \implies [r_n, r] = 0$. Using a cascading argument, $\forall i = 0, 1, \cdots, , n, [r_n, r_i] = 0$. Now inductively show that $[r_{n-1}, r_i], [r_{n-2}, r_i]$ etc are all zero.

Observe that from the properties of Jacobson radical, $1 - rt$ is an invertible element in $R[t]$. From Lemma 3.8, r_n is a nilpotent element. Thus I is further a non-zero nil ideal of R. □

Corollary 3.41. *Let $\{R_\alpha : \alpha \in \mathcal{A}\}$ be a collection of prime PI-rings. Then, each $R_\alpha[t]$ is a semiprimitive ring.*

Proof. Follows from Theorem 3.8 and Theorem 4.2. □

Chapter 4

Invertibility of Non-Commutative Rational Functions

In this section, we prove an important fact about non-commutative rational expressions:

Theorem 4.1. *Any non-zero non-commutative rational expression in $d \times d$ generic matrices must be invertible.*

This uses a theorem of Amitsur that states that Universal Division Algebras (to be defined later) are in fact division algebras in the usual sense.

We largely follow the proof presented in Visu Makam's thesis. We first present some preliminaries; the remaining material needed is present in Chapter 2

Let K be an infinite field.

4.1 A Localization Theorem

In this section, we prove a localization theorem due to Posner. In the commutative setting, every integral domain has a *field of fractions*—a smallest field in which the domain embeds. Posner seeks to generalize this construction to the non-commutative setting.

Informally, he shows that a prime PI-ring R admits a quotient ring $Q(R)$ that is also a *primitive* PI-Ring that satisfies the same polynomial identities as R. We present a

proof based on *central polynomials*, avoiding the use of *Goldie's theorem*, which appears in Posner's original argument.

Lemma 4.2. *Let R be a prime ring with a center C. Let $S = C \setminus \{0\}$. Then*

1. *C is a domain*
2. *$S^{-1}C$ is the center of $S^{-1}R$*

Proof. 1. Let $a, b \in C$ be non-zero. We want to show that $ab \neq 0$. $\forall r \in R, ar = ra, br = rb$. Also, R is prime. For any $x, y \in R$, if $xRy = \{0\}$, then $x = 0$ or $y = 0$. Now consider $aRb \subseteq R$. Then,

$$aRb = \{arb : r \in R\} = \{abr : r \in R\}$$

So if $ab = 0$, then $aRb = \{0\}$, contradicting the fact that R is a prime ring.

2. We show both directions.
$\implies$ $S^{-1}C \subseteq Z(S^{-1}R)$
Let $\frac{c}{a} \in S^{-1}C$ with $c \in C, s \in S$ and $\frac{r}{t} \in S^{-1}R$ with $r \in R, t \in S$
Then, $\frac{c}{s} \cdot \frac{r}{t} = \frac{cr}{st} = \frac{rc}{st} = \frac{r}{t} \cdot \frac{c}{s}$. Thus $\frac{c}{s} \in Z(S^{-1}R)$
To show $Z(S^{-1}R) \subseteq S^{-1}C$, suppose $z \in Z(S^{-1}R)$. Then $\forall\ \frac{r}{s} \in S^{-1}R$, $z \cdot \frac{r}{s} = \frac{r}{s} \cdot z$ holds. Suppose $z = \frac{r_0}{s_0}$ for some $r_0 \in R, s_0 \in S$. We have to show $\frac{r_0}{s_0} \in S^{-1}C$, that is $r_0 \in C$. Let $r \in R$. Then, $\frac{r_0}{s_0} = \frac{r_0}{s_0} \cdot \frac{r}{1} = \frac{r}{1} \cdot \frac{r_0}{s_0} \implies \frac{r_0 r}{s_0} = \frac{r r_0}{s_0}$ and thus $r_0 \in C$.
Finally, $z = \frac{r_0}{s_0} \in S^{-1}C$

□

Lemma 4.3. *Suppose R is semiprime ring and t is a central indeterminate. Then $R[t]$ is a semiprimitive Pi-ring*

Proof. By Corollary 3.9, $R[t]$ is a PI-ring. Let $R_\alpha : \alpha \in \mathcal{A}$ be a collection of prime PI-rings. Then the following map induces a subdirect product:

$$R \to \prod\{R_\alpha : \alpha \in \mathcal{A}\}$$

By Corollary 3.41, each $R_\alpha[t]$ is semiprimitive. Then $R[t] \to \prod\{R_{\alpha[t]} : \alpha \in \mathcal{A}\}$ expresses $R[t]$ as a subdirect product of semiprimitive rings. □

The following lemma is due to Rowen [Row74].

Lemma 4.4. *Let R be a semiprime PI-ring with a center C. Suppose J is a non-zero two sided ideal of R. Then $J \cap C \neq 0$.*

Proof. Recall that a ring is said to be semiprimitive if its Jacobson radical is zero. Introduce a central indeterminate t, and consider the polynomial ring $R[t]$. By Lemma 4.1, $R[t]$ is a semiprimitive PI-ring. The center of $R[t]$ is $C[t]$. Further for any ideal $J \subseteq R$, $(J[t] \cap C[t]) = (J \cap C)[t]$. This essentially means studying the question for R and J is equivalent to studying the question for $R[t]$ and $J[t]$. Hence we can now assume R to be semiprimitive.

Since R is semiprimitive and satisfies a PI of degree d, we can write it as its subdirect product decomposition of primitive PI rings R_α.

$$R \hookrightarrow \prod_{\alpha \in \mathcal{A}} R_\alpha$$

Let n_α be the PI-degree of R_α. By Kaplansky's theorem, each R_α can be seen as a centrally simple ring of dimension n_α^2 over its center. Further, $2n_\alpha \leq d$, as R_α cannot satisfy any polynomial identity of degree $< 2n_\alpha$ by Lemma 3.22. So the collection $\{n_\alpha\}$ is bounded above.

We now understand how J maps under this decomposition. $J \subseteq R \hookrightarrow \prod R_\alpha$, and its projection to R_α is either 0 or all of R_α due to primitivity. But since $J \neq 0$, at least one projection is non zero, and thus for some α, $J \mapsto R_\alpha$.

Let n be the maximum n_α for which the projection of J onto R_α is all of R_α. We make use of the Formanek polynomial introduced earlier, which Is a central polynomial for all central simple algebras of PI-degree n. We now plug elements of J into $F_n =: f$ and see what happens in each R_α.

- If $n_\alpha > n$, the projection is 0. Thus $f(J) = 0$
- If $n_\alpha < n$, then F_n is a PI for R_α, by a theorem of Procesi.

- If $n_\alpha = n$, since F_n is a central polynomial for R_α, it evaluate to a central and maybe nonzero element in R_α.

From these observations, $F_n(J) \subseteq J \cap C$ contains some non-zero element, as required. □

Lemma 4.5. *Let R be a semiprime PI-ring with center C. If C is a field, then R is a finite dimensional central simple C-algebra*

Proof. By Lemma 4.4, R is simple and thus primitive. Using Theorem 3.21, R is a finite dimensional simple algebra over its center C. □

Theorem 4.6. *Let R be a prime PI-ring with a center C. Let $S = C \setminus \{0\}$. Then,*

1. *$S^{-1}R$ is a central simple $S^{-1}C$ algebra.*
2. *R and $S^{-1}R$ satisfy the same polynomial identities as $K\langle X\rangle$*

Proof. From Lemma 4.2, $S^{-1}R$ is a prime ring and its center $S^{-1}C$ is a field. For (2), we consider two cases:

- Suppose C is finite. Then C is a field, and thus $R = S^{-1}R$.
- Suppose C is infinite. Then from Corollary 3.9, R and $R[T]$ satisfy the same set of polynomial identities for any set of commuting indeterminates T. $S^{-1}R$ is a homomorphic image of $R[T]$ for large enough T, and thus $S^{-1}R$ also satisfies every identity satisfied by R. The converse follows because $R \subseteq S^{-1}R$.

□

Corollary 4.7 (Posner)**.** *The central quotient of a prime Polynomial Identity Ring is a simple algebra.*

Proof. $S^{-1}R$ is a prime PI-ring whose center is a field. By Lemma 4.1, it is also a finite dimensional central $S^{-1}C$ algebra. □

4.2 A theorem of Amitsur

Theorem 4.8 (Amitsur)**.** *The ring of $d \times d$ generic matrices R_d is an non-commutative integral domain.*

Proof. For the sake of this proof, we use $K\{X\}$ to denote the ring of generic matrices. Let $X_1, \cdots$, be $n \times n$ generic matrices. Let $R = K[\{x^i_{jk}\}]$ be the polynomial ring over these variables. As seen earlier, $K\{X\} \subseteq M_n(R)$.

Now consider $K\{X\} \subseteq M_n(E)$, where $E = K(\{x^i_{jk}\})$, where E is the quotient field of R. Thus $K\{X\}$ can be viewed as a subalgebra of $M_n(E)$.

We use a linear independence argument.

Claim: The n^2 matrices $\{X_1^i X_2 X_1^j : i \leq i, j \leq n\}$ are linearly independent over E, and thus form a basis for $M_n(E)$.

To show this, we consider the $n^2 \times n^2$ matrix formed by flattening each $X_1^i X_2 X_1^j$ into a $1 \times n^2$ matrix, and then "stacking" them.

The determinant of this matrix is a polynomial in the variables of X_1 and X_2. Our goal is to show this is non-zero. To do this, we make X_1 a diagonal matrix by setting all its off diagonal entries to non-zero.

Now, suppose under this setting $X_1 = \text{diag}(t_1, \cdots, t_n)$ where the t_i's are new algebraically independent variables. Then the entries of $X_1^i X_2 X_1^j$ form a Vandermonde style matrix, and thus its determiant is a non-zero polynomial.

Therefore, $K\{X\}$ contains an E-basis for $M_n(E)$, and is thus a prime PI ring by Lemma 3.16.

Suppose for contradiction, $K\{X\}$ is not a domain. Because it is a prime ring, it has a non-zero element whose square is zero, by Lemma 3.38. This then corresponds to a non identity $f \in K\langle X\rangle$ that vanishes in $K\{X\}$ (due to the isomorphism by Theorem 3.26) and thus lies in the t-ideal of polynomial identities of $M_n(K)$, denote this by $M(n)$. But f^2 is an element of $M(n)$.

We now use the domain construction from section . Let $R = (L, \phi, \sigma^{\pm 1}$ be the domain under consideration. By Corollary 4.7, R has the same identities satisfied by $M_n(K)$.

Since R is a domain, and f would be a PI for R but f^2 isn't a PI for R. This is absurd as R is a domain.

□

Corollary 4.9. *R_d is a prime ring.*

Proof. Every non-commutative integral domain is a prime ring. □

4.3 Tying it all up

Finally, we present the main theorem of this section:

Let Z_d denote the center of R_d. Recall the center of a ring R is a subring with all elements x such that $xy = yx \forall y \in R$. Let the field of fractions of Z_d be denoted by Q_d.

Definition 4.10 (Central Localization)**.** $UD(d) := Q_d \otimes_{Z_d} R_d$

$UD(d)$ is called the *central quotient*. Amitsur showed that this is infact a division algebra, and is called a universal division algebra of *degree* d.

Lemma 4.11. *$UD(d)$ has no non-zero nilpotents.*

Proof. Elements of $UD(d)$ can be seen as Q_d-linear combinations of elements $1 \otimes r$ where $r \in R_d$. Every element $x \in UD(d)$ can be written as

$$x = \sum_{i=1}^{k} q_1 \otimes r_i$$

where $q_i \in Q_d$ and $r_i \in R_d$. Now each q_i can be written as $q_i = z_i^{-1} z_i'$ for some $z_i, z_i' \in Z_d$, with $z_i \neq 0$. Thus,

$$x = \sum_{i=1}^{k} (z_i^{-1} z_i') \otimes r_i = \sum_{i=1}^{k} 1 \otimes (z_i^{-1} z_i' r_i)$$

Now elements of $UD(d)$ are sums of elements which are of the form $1 \otimes (z^{-1} r)$ where $z \in Z_d \setminus \{0\}, r \in R_d$. But R_d is localized at $S = Z_d \setminus \{0\}$, and thus the element r/z corresponds to $1 \otimes (z^{-1} r)$.

Suppose $x = /z$ is a nilpotent. $\exists n > 0$ such that $(r/z)^n = 0$ in $UD(d)$. Since z is central, $(r/z)^n = \frac{r^n}{z^n}$ and $\frac{r^n}{z^n} = 0 \implies r^n = 0$ in R_d. But R_d is an integral domain! So we have $r = 0 \implies r/z = 0 \implies x = 0$. □

Theorem 4.12 (Universal Division Algebras are division algebras)**.** *$UD(d) := Q_d \otimes_{Z_d} R_d$ is a division algebra and is called a universal division algebra of degree d.*

Proof. By Corollary 4.2, R_d satisfies the Amitsur-Levitzki polynomial, that is s_{2d}, by Theorem 3.14 and is thus a polynomial identity ring. Further, it is a domain and thus a prime ring. By Corollary 4.7, the central quotient of a prime PI-ring is a simple algebra, and thus $UD(d)$ is a simple algebra. From Theorem 3.17, $\exists r \in \mathbb{N}$ and a division algebra D such that $UD(d) \cong M_r(D)$. From Lemma 4.3, $UD(d)$ has no non-zero nilpotents. Finally, $UD(D) \cong M_1(D) \cong D$, finishing the proof. □

Corollary 4.13. *Any non-zero non-commutative rational expression in $d \times d$ generic matrices must be invertible.*

Proof. $UD(d) \subseteq M_d(K\{x^i_{jk}\})$. Informally, R_d lives inside matrices over polynomials. After localization, we are left with matrices over rational functions. □

Chapter 5

Polynomial Identity testing

In this chapter, we introduce Polynomial Identity Testing, as well as a few other notations and concepts that come of use later.

5.1 Polynomial and Matrix Identities

Let $\mathbb{F}$ be a field, and A be an associative A-algebra, particularly the algebra $\text{Mat}_d(\mathbb{F})$ of $d \times d$ dimensional matrices over $\mathbb{F}$.

$\mathbb{F}[x_1, x_2, \cdots, x_n]$ denotes the ring of commutative polynomials polynomials with coefficients in $\mathbb{F}$ over n variables. Thus every polynomial is a linear combination of monomials, where each monomial is a product of variables. We can now define a non-commutative polynomial over $\mathbb{F}$ as a formal linear combination of monomials where the product of variables do not commute: these are polynomials over the free algebra.

A polynomial $f(x_1, x_2, \cdots x_n) \in$ is an identity for the algebra A, if $\forall c \in A, f(c) = 0$. In our case, A is a matrix algebra, and f is called a matrix identity. A matrix identity (for a particular matrix algebra) can thus be seen as a non-commutative polynomial that vanishes over all assignments of matrices to variables. As seen earlier, the standard identity in $2n$ variables is the polynomial identity for the $n \times n$ matrix algebra.

5.2 Polynomial Identity Testing

Polynomial Identity Testing (PIT) is the problem of efficiently determining whether a given polynomial is identically the zero polynomial. We are interested in testing PIT using arithmetic circuits: Given an arithmetic circuit $\mathcal{C}$ computing a polynomial p, decide whether $\mathcal{C}$ computes the zero polynomial or not. Note that we are interested in deciding only whether the polynomial is identically 0; not whether the polynomial evaluates to the zero polynomial over the field: for example, $x^2 - x \in \mathbb{F}_2[x]$ is not a zero-polynomial, but $xy - yx \in \mathbb{F}[x_1, x_2, \cdots x_n]$ is. For infinite fields, one can see that a polynomial evaluates to zero over all elements of the field if and only if the polynomial is identically zero (use induction and the fact that any non-zero univariate polynomial of degree d over a field $\mathbb{F}$ has at most d roots.

Suppose the input polynomial is given as a list of coefficients (and the corresponding monomials), the problem is simple: check whether the list has a non-zero entry or not. When the input polynomial is given as a circuit, however, expanding the polynomial out and checking whether there is a non-zero coefficient is a costly affair: you can have an exponential (in the commutative case) or a doubly exponential (in the non-commutative case) number of monomials.

Before delving deeper, we shall state the problem formally.

Let $C_{s,d}$ be the set of algebraic circuits of size $\leq s$ and degree $\leq d$ computing polynomials in $\mathbb{F}[x_1, x_2, \cdots x_n]$. Let $\phi \neq \mathfrak{C} \in C_{s,d}$.

Problem 5.1 (PIT)**.** *Given a circuit $C \in \mathcal{C}$, computing a polynomial f, determine whether $f \equiv 0$*

We usually try to find an efficient algorithm that does in poly(s, n, d) steps. PIT can either be blackbox or whitebox.

Whitebox PIT: Here the polynomial is given to us as a circuit and we have access to all the gates inside the circuit. Some classes of circuits have trivial whitebox PIT algorithms. For instance,

Blackbox PIT: Here, we are not allowed to "look into" the arithmetic circuit: that is, we only have oracle access to the circuit and can evaluate it at certain points (field

elements or matrices, according to whether we are in the commutative or non-commutative case respectively). We are in search of an efficient PIT algorithm in its input parameters, usually the size of the circuit. Blackbox PIT is equivalent to the concept of hitting sets. Again, let us define this notion formally:

Let $C_{s,d}$ be the set of algebraic circuits of size $\leq s$ and degree $\leq d$ computing polynomials in $\mathbb{F}[x_1, x_2, \cdots x_n]$. Let $\phi \neq \mathfrak{C} \in C_{s,d}$. A hitting set $\mathcal{H} \subseteq \mathbb{F}^n$ for $C_{s,d}$ is a set of points such that if $C \in C_{s,d}$ computes some non-zero polynomial f_C, then $\exists \bar{a} \in \mathcal{H}$ such that $f(\bar{a}) \neq 0$.

The converse follows trivially; if a circuit C computes the zero polynomial f, then $\forall \bar{a} \in \mathcal{H}$, $f(\bar{a}) = 0$

Thus, a poly(s, n, d) hitting set $\mathcal{H}$ for a set of circuits $\mathcal{C}$ amounts to giving an efficient blackbox PIT algorithm for $\mathcal{C}$. We just evaluate f_C on all points of $\mathcal{H}$. If $\exists \bar{a} \in \mathcal{H}$ such that $f_C(\bar{a}) \neq 0$, we can output non-zero. Otherwise output zero. This takes no more than $O(s)$ field operations.

The use of arithmetic circuits is natural: for there are $2^{\binom{n+d}{d}}$ possible monomials for a degree-d polynomial over $\mathbb{F}\langle x_1, x_2, \cdots x_n \rangle$. Simply expanding and checking whether coefficients cancel out would be infeasible. Even in the commutative case, the trivial way of expanding out is inefficient: there are $\binom{n+d}{d}$ possible monomials. From now on, we will clearly distinguish whether we are referring to commutative PIT (c-PIT) or non-commutative PIT (nc-PIT).

The question of c-PIT is one of the central problems of algebraic complexity theory. The proofs of several important theorems like the AKS Primality Test [AKS04], IP=PSPACE use PIT techniques.

c-PIT has no efficient deterministic algorithm, Indeed, derandomizing c-PIT has great consequences for arithmetic circuit and boolean circuit lower bounds due to a result of [KI04]

5.3 Polynomial Identity Lemma

c-PIT, however, has a co-RP algorithm due to Schwartz and Zippel. Due to its vast history, we call it the Polynomial Identity Lemma. We state it here:

Theorem 5.2 (Polynomial Identity Lemma)**.** *Let $f(x) \in \mathbb{F}[x]$ be a non-zero, n-variate degree d polynomial. Let $S \subseteq \mathbb{F}$ be a set of size $> d$. If $a_1, a_2, \cdots a_n$ are chosen independently and uniformly at random from S, then*

$$\Pr_{(a_1,a_2,\cdots a_n)}[f(a_1, a_2, \cdots a_n) = 0] \leq \frac{d}{|S|}$$

Proof Idea: We use the fact that a univariate polynomial of degree d has atmost d distinct roots, and then induct on the number of variables.

Proof. We prove this by induction on the number of variables n.

Base case: $n = 1$
Let $f(x)$ be a non-zero univariate polynomial of degree at most d. Then f has at most d roots. Thus, if $a \in S$ is chosen uniformly at random, $\Pr_a[P(a) = 0] \leq \frac{d}{|S|}$.

For the induction step, assume the lemma holds for $n-1$ variate polynomials. We will prove it for n variate polynomials. Let $g(x_1, x_2, \cdots x_{n-1})$ be any non-zero polynomial over $n-1$ variables of degree at most d. Then, by the hypothesis,

$$\Pr_{(a_1,a_2,\cdots a_{n-1})}[g(a_1, a_2, \cdots a_{n-1}) = 0] \leq \frac{d}{|S|}$$

Now, let $f(x_1, x_2, \cdots x_n$ be an n-variate polynomial of degree at most d. Rewrite f as follows:

$$f(x_1, x_2, \cdots x_n) = \sum_{i=0}^{d} f_i(x_1, x_2, \cdots x_{n-1}) \cdot x_n^i$$

where each f_i is a polynomial in $n-1$ variables. Since f is not identically zero, all the f_i's cannot be identically zero simultaneously. Thus there is atleast one polynomial f_i with non zero coefficients. Let i_0 be the largest i suchh that $f_{i_0} \neq 0$. Then,

$$\begin{aligned}
\Pr(\bar{a} = 0) &= \Pr[P_i(\bar{a}) = 0] \cdot \Pr[P(\bar{a}) = 0 | P_i(\bar{a}) = 0] + \Pr[P_i(\bar{a}) = 0] \cdot \Pr[P(\bar{a}) = 0 | P_i(\bar{a}) \neq 0] \\
&\leq \frac{d - i_0}{|S|} + \frac{i_0}{|S|} \\
&\leq \frac{d}{|S|}
\end{aligned} \tag{5.1}$$

□

Observe that the algorithm given the PI lemma is a *blackbox* algorithm: we only needed evaluation points as input to the circuit, and not the structure of the circuit. We also have *whitebox* PIT algorithms, which exploit the structure of the circuit in some way. More formally:

5.4 Applications of PIT

PIT has a wide range of applications, both in the design of algorithms for problems that may seem to be unrelated at first g;ace, as well as for establishing lower bounds in computational complexity. We highlight a few examples below:

- **Primality Testing:** The first deterministic polynomial-time primality testing algorithm, the AKS algorithm [AKS04] was developed by formulating primality as a special instance of PIT: testing whether $Ps(x) = (x+1)^n - (x^n+1)$ over $\mathbb{Z}/n\mathbb{Z}$. This resolved an important open question in computation.
- **Perfect Matching**: A much older application is in perfect matching due to Tutte's theorem that states that a graph has no perfect matching if and only if the determinant of its Tutte matrix A is zero. This involves checking whether $\det A = 0$, which becomes a special case of PIT.
- **Polynomial Equivalence Testing:** Determining whether two polynomials f and g are identical reduces to checking if $f - g \equiv 0$. This reduction is crucial in various models, such as verifying the equivalence of two read-once oblivious algebraic branching programs (ROABPs).
- **Circuit Lower Bounds:** PIT is deeply intertwined with circuit complexity. In a celebrated result, Kabanets and Impagliazzo [KI04] showed that a deterministic PIT algorithm would imply either a separation in boolean circuit complexity classes (NP $\not\subseteq$ P/poly) or a separation in algebraic circuit complexity classes (VP $\neq$ VNP).
- **Polynomial Factorization:** It has been shown that deterministic multivariate polynomial factorization reduces to deterministic PIT [KSS14]

- **IP=PSPACE** Testing whether two multivariate polynomials are equal by evaluating them at randomly chosen points played a key role in proving IP=PSPACE [Sha92]

Thus there is no doubt that PIT is an interesting as well as important problem to be studied.

5.5 Derandomizing PIT

Using the Polynomial Identity Lemma, we obtain a simple randomized algorithm for solving blackbox PIT. major research direction has been the effort to derandomize PIT: that is, to design an efficient deterministic algorithm for the problem. One approach to derandomization is to systematically break down the general PIT problem into smaller, more manageable subproblems, by studying PIT for restricted classes of circuits (such as formulas and algebraic branching programs (ABPs), as we have already seen). Similar efforts have also been made in the non-commutative setting.

In this thesis, however, we focus solely on developing an efficient randomized algorithm for PIT in the setting of non-commutative arithmetic circuits with potentially exponential degree.

Chapter 6

Non-Commutative Polynomial Identity Testing: A Survey

In the following sections, we will survey relevant results (both recent, and ancient) for nc-PIT.

6.1 Whitebox ABP

In an attempt to prove lower bounds in the non-commutative model, Nisan introduced the ABP model. In [RS04], Raz and Shpilka gave a deterministic polynomial time whitebox algorithm for PIT of non-commutative formulas. They first efficiently simulated the formula using an ABP; then gave an algorithm for PIT of homogeneous ABPs. ABPs can be homogenized with only a polynomial blowup.

Proof Idea:

Simulation of Non-Commutative Formulas by ABPs

We first create a temporary ABP. For each gate ϕ in the formula, we construct the ABP accordingly.

- When ϕ is an input gate,create a trivial ABP with one edge labelled accordingly.

- When $\phi = \phi_1 \times \phi_2$, sequentially connect the ABPs for ϕ_1 and ϕ_2 (make the sink of ABP(ϕ_1) the source for ABP(ϕ_2)
- When $\phi = \phi_1 + \phi_2$, make a parallel connection: both ABPs start from the same source and sink and reach the same target.

We now make this temporary ABP layered by degree such that the vertices at layer i correspond to homogeneous polynomials of degree i. For each vertex in the temporary ABP

- Create r copies $(v, 1), (v, 2), \ldots, (v, r)$ where $r = \deg \phi$. Place all copies (v, i) in the layer i
- Rewrite the edges to preserve the degree. If an edge from u to v in the previous ABP contributed some degree d, then in the final ABP connect (u, i) to $(v, i + d)$. This may incur a size blowup of almost $O(r)$.

We now present the idea for the ABP PIT.The idea comes from Nisan's rank bound using the Partial Derivative Matrix.

Let $\mathcal{C}$ be an arithmetic circuit computing the polynomial f. For every degree 2 monomial of f, say $x_i x_j$, construct a new ABP that computes the portion of f whose monomials begin with $x_i x_j$. Then, $\mathcal{C}$ computes the zero polynomial if and only if each of the individual ABPs compute the zero polynomial (we are in a non-commutative setting). To do this, a depth reduction is performed that merges all the new ABPs to a single ABP, say $\tilde{C}$ with one level lesser than $\mathcal{C}$. The important trick here is to relabel the edges accordingly: simply introducing a new variable y_{ij} for $x_i x_j$ would not work because it could result in exponentially many new variables.

Thus, they fully expand the ABP in the monomial one variable at a time. Then, for each new variable, they change the basis to reduce to the rank bound.

6.2 Non-Commutative Randomized Blackbox PIT

Bogdanov and Wee [BW05] gave a randomized polynomial time-algorithm for *non-commutative* arithmetic circuits where the degree of the circuit is bounded by the size. They use a classical theorem loaned from the study of Polynomial Identity Algebras, called the **Amitsur-Levitzki** theorem [AL50].

Theorem 6.1. *Non-Commutative Randomized Blackbox PIT Let $\mathbb{F}$ be a field. Let X be a set of indeterminates. Any non-zero polynomial $p \in \mathbb{F}\langle Z\rangle$ of degree $\leq 2d-1$ is not a polynomial identity for the matrix algebra $\mathbb{M}_d(\mathbb{F})$*

In other words, P does not vanish on all $d \times d$ matrices over $\mathbb{F}$.

Proof. Suppose $\mathbb{M}_d(\mathbb{F})$ satisfies a polynomial identity of degree $k < 2d$. It must be that it satisfies a multilinear identity of the following form:

$$f(x_1, x_2, \cdots, x_k) = \sum_{\sigma \in S_k} \alpha_\sigma x_{\sigma(1)} \cdots x_{\sigma(k)}$$

where $\alpha_\sigma \in \mathbb{F}$.(This follows from a linearization argument). Let $\{e_{ij}\}_{i,j=1}^{k}$ be the basis of the $k \times k$ matrix algebra. We then use staircase arguments:

$$f(e_{11}, e_{12}, e_{22}, e_{23} \cdots e_m n) = \alpha_\varepsilon e_{1m}$$

. Depending on whether k is even or odd, $m = n$ or $m = n-1$, and ε is the identity permutation. It follows that $\alpha_\epsilon = 0$. □

They then used the above theorem to obtain a randomized PIT algorithm.

Instead of field elements like in The PI lemma, they substitute elements from the $d \times d$ matrix algebra (technically, matrices over the field extension) to test PIT for circuits computing polynomials of syntactic degree $\leq 2d-1$. We state a non-commutative analogue of Schwarz Zippel Lemma:

Theorem 6.2 (Non-Commutative Schwartz Zippel)**.** *Let $f \in \mathbb{F}\langle x_1, x_2, \cdots x_n$ be a non-commutative polynomial of degree d. Let $\mathbb{L}$ be a field extension of F. Let $\{e_{ij}\}, 1 \leq i, j \leq k$ be the standard basis of $\mathbb{M}_k(\mathbb{F})$, L_k be a linear subspace of $\mathbb{M}_k(\mathbb{L})$ generated by some subset of the unit matrices, and $T \subseteq \mathbb{L}$. Then, one of the two holds:*

- *f is a Polynomial Identity for* L_k
- $\Pr[f(M_1, M_2, \cdots M_n) = 0] \leq \frac{d}{|T|}$

Proof. Choose a basis $e_1, \ldots, e_{k^2}$ for $\mathbb{L}^{k\times k}$ such that the $\mathbb{L}^k$ is generated by $e_1, \ldots, e_l$. Then define a polynomial map that expresses each x_i as follows:

$$x_i = \sum_{j=1}^{l} y_{ij} e_j$$

where y_{ij} are new indeterminates. Under this map, the polynomial f is transformed into a polynomial $q \in \mathbb{F}\langle y_{11}, \ldots, y_{n]}\rangle$ whose evaluations represent $k \times k$ matrices. Further, q has a unique decomposition like

$$q(y_{11}, \cdots y_{nl}) = \sum_{j=1}^{k^2} q_j(y_{11}, \cdots y_{nl}).$$

Each q_j has degree almost d, and $q \equiv 0$ if and only if all the q_j vanish identically. Let i_0 be the largest index such that $q_{i_0} \equiv 0$.

Choose random matrices $M_1, \cdots M_n$ to keep $q_{i_0}(a_{11}, \cdots a_{nl})$ non-zero, and by the polynomial map, $f(M_1, \cdots M_n) \neq 0$.

□

From this theorem follows a black-box, randomized nc-PIT algorithm for circuits computing polynomials with degree bounded by size.

Theorem 6.3. *Non-Commutative Randomized Blackbox PIT There is a randomized black-box PIT algorithm for circuits computing n-variate polynomials of degree atmost d.*

Proof Idea: Let $\mathcal{C}(x_1, x_2, \cdots x_n)$ be a nc-circuit of syntactic degree $\leq 2d - 1$ For each $i \in [n]$, substitute x_i by a matrix M_i with commuting entries. $M_p = p(M_1, M_2, \cdots M_p)$ is not identically zero. Then, some entry in M_p has a commutative non-zero polynomial, say g. But $\deg(g) < 2d$. We then randomly susbtitute elements from a field of size $4d$: Using the non-commutative Schwarz Zippel Lemma seen above, we get a randomized PIT with error $\leq \frac{1}{2}$.

However, this approach cannot be used for an efficient randomized algorithm for general non-commutative circuits, as the dimension of the matrices needed grows linearly with the degree of the polynomial. The following is a well-known open problem in the community:

Open Problem: WACT 2016

Let $f \neq 0$ be a non-zero polynomial computed by a non-commutative circuit of size s and degree 2^s. Then, there exist matrices $B_1, B_2, \cdots, B_n$ of dimension poly(s) such that $f(B_1, B_2, \cdots B_n) \neq 0$

The goal of this thesis is to move closer towards solving this problem.

Bogdanov and Wee [BW05]also gave some important query complexity lower bounds for non-commutative circuits. These lower bounds imply that an improvement to the blackbox algorithm for general circuits (with unbounded degree) either needs simulating an algebra of exponential dimension that somehow can he represented implicitly (because otherwise, our model of computation would not have enough space for storing the matrices) or needs to exploit properties of the circuits themselves. This observation motivated us to look into properties of the circuits, and PIT algorithms for other restricted classes that have developed over the years.

6.3 nc-PIT for sparse polynomials

Arvind, Mukhopadhyay and Srinivasan introduced the idea of using automata theory to design efficient nc-PIT algorithms. This is a natural choice, because our monomials can be seen as words over the free algebra. In [AMS08], they gave an efficient deterministic whitebox PIT algorithm for non-commutative circuits computing n-variate polynomials of degree d with t monomials. This algorithm ran in poly$(d, n, |C|, t)$.

We are more interested in another randomized algorithm by Arvind, Joglekar, Mukhopadhyay and Raja [Arv+19].

In [Arv+19], they show the following theorem about non-commutative identities:

Theorem 6.4. *Theorem LLet $\mathbb{F}$ be a field of size $\geq (n+2)d$. Let $f \in \mathbb{F}\langle x_1, x_2, \cdots x_n\rangle$ be a non-zero degree d polynomial of sparsity t. Then, f cannot be a polynomial identity for the $k \times k$ matrix algebra where $k \geq \lceil \log t \rceil + 1$*

They prove this theorem using techniques from automata theory. We briefly detail the idea below:

Proof Idea:

- We first compute the non-commutative polynomial to a polynomial over $\mathbb{F}\langle x, y\rangle$ using a simple encoding trick. This is a bijective encoding, and thus it is enough to study non-commutative polynomials over $\mathbb{F}\langle x, y\rangle$.
- For every subset of monomials, construct an *isolating index set*. Let $\mathcal{M}$ be a subset of degree d monomials. $\mathcal{I} \subseteq [d]$ is an isolating index set for $\mathcal{M}$ if $\exists m \in \mathcal{M}$ such that $\forall m' \neq m, \exists i \in \mathcal{I}$ such that $m'[i] \neq m[i]$. In simple terms, it is the collection of all those indices where the monomials differ in atleasr one index.
- We then show that every subset of monomials has an isolating index set of size $k \leq \log |\mathcal{M}|$, using a halving argument.
- Construct a *small* substitution automata (an NFA) to guess the isolating index set for the monomials in the support. Even though there are (more than) exponentially many wrong guesses, using a substitution automata alleviates this issue because the monomials computed on different paths have disjoint support.
- Thus a nonzero polynomial is output by the automata (the isolated monomial can never get cancelled)

From the above theorem, a randomized blackbox PIT follows using a Schwarz Zippel type argument (substitute $(\log t + 1) \times (\log t + 1)$ matrices).

We get a $\text{poly}(\log d, n, \log t)$ randomized PIT. However, this also becomes exponential in case we have doubly exponential monomials in the support.

6.4 PIT for UPT circuits

Lagarde et. al in [LLS18] gave deterministic whitebox PIT algorithms for arithmetic circuits of bounded degree with unique parse trees. They also gave PIT for sum of

constantly many UPTs. Their idea is an adaptation of Raz and Shpilka's algorithm for whitebox ABP.

We tried to adapt this work to the case we are interested in: homogeneous arithmetic circuits of exponential degree. Their deterministic whitebox PIT algorithm, inspired by Rank Bound ideas of Raz and Shpilka works in the case of exponential degree, on the condition that the parse trees of the monomials have small size. This is because we are inducting on the size of the parse tree. Thus, we don't get an efficient PIT algorithm for circuits with unrestricted degree.

6.5 PIT for sum of UPT circuits

They also gave whitebox PIT for sum of k many UPT circuits by modifying an algorithm for ROABPs by [Gur+17]. This is equivalent to the problem of equivalent testing of two UPTs, say P and Q, and this is the problem we will be tackling.

Theorem 6.5. *Let $N, k, s \in \mathbb{N}$ be some parameters. Then there is a deterministic $s^{O(2^k)}$ algorithm, which on input $k+1$ UPT circuits $C_0, C_1, \ldots, C_k$ each of size $\leq s$ over N variables tests whether the circuit computed by their sum computes the zero polynomial.*

Proof Idea:

- Transform each circuit C_i to its normal form. This transformation can incur a quadratic blowup. We can also assume that each T_i has fan-in ≤ 2.
- Construct a set of characterizing identities for the polynomial P of size $poly(N, s, d)$.
- Verify whether these identities are satisfied by Q. This is done by a call to the algorithm in time $s^{O(2^k)}$.
- Then show that if Q satisfies these identities, and if P and Q agree on even a small set of coefficients, P and Q are identical. Comparing these coefficients is also done in $poly(s)$ time.

It works efficiently for the first few steps, but the final step is to test the equality of coefficients of P and Q using an automata construction of [AMS08]. Their automata takes time poly(s, d) to construct; this is inefficient for our use case.

Chapter 7

Rank Concentration

The idea of rank concentration was first introduced in the context of developing black-box PIT algorithms for set-multilinear circuits , as discussed in Chapter 2.

An algebra A is a vector space V along with a vector product. The vector product is a binary operation from $V \times V$ to V, and is distributive and associative with respect to the addition operation of the v. sp. We have already learnt discussed

When the vector product is a co-ordinate wise product it is called a **Hadamard Algebra**. A k-dimensional hadamard algebra is denoted by $\mathbb{H}_k$.

Definition 7.1 (Rank Concentration)**.** A polynomial $f(x)$ over an algebra A is said to be *ℓ-concentrated* if the coefficients of its $< \ell$ support monomials span all of its coefficients.

If a polynomial f is ℓ-concentrated, then evaluating it on the unit cube suffices to receiver all its coefficients via linear combinations. However, a general polynomial g need not be ℓ-concentrated.

Agrawal, Saha and Saxena [ASS13] introduced a variable shift technique to show concentration. For a general polynomial over $\mathbb{H}_k$, applying a formal shift $x_i \mapsto x_i + t_i$, where t_i are formal variables results in the shifted polynomial being $O(\log k)$ concentrated over $\mathbb{F}(t_1, \ldots, t_n)$, provided the number of monomials is small. But since the monomials are usually exponential in number, the shift is applied only to the low-degree parts of the circuit. For example, consider the product $P = \prod_{i=1}^{m} P_i(x_i)$. The shift ensures every

subproduct of $l = O(\log k)$ is ℓ-concentrated. Finally, they argue inductively that this local concentration implies global concentration, that is concentration of the full product

7.1 Basis Isolation

For the sake of completion, we also provide an overview of the Basis Isolation Weight Assignment (BIWA) idea introduced by [Agr+14]. This was an improvement to the rank concentration idea studied earlier, and is used for the construction of hitting sets of ROABPs. As seen earlier, ROABPs can be written as IMM products. That is, the output polynomial can be expressed as some

$$f(x) = c^{\intercal}(\textstyle\prod_{i=1}^{d} F_i(x))d$$

where each $F_i(x)$ is a matrix with polynomial entries. The key idea is to view $F(x) := \prod_{i=1}^{d} F_i(x)$ as a polynomial over the matrix algebra, say $\mathbb{F}^{w\times w}$ and to study its coefficient space. It then suffices to evaluate $D(x)$ on a set of points that span its coefficient space.

The BIWA technique aims to isolate a basis for the coefficient space of $F(x)$ by assigning weights to the variables such that the coefficients of a small set of monomials, those with minimum weight form a basis. Such an assignment ensures that the basis monomials receive distinct weights, and every other coefficient lies in the span of those smaller weights. This guarantees that the substituted polynomial is nonzero provided the input polynomial is.

Chapter 8

Rank Concentration in Non-Commutative Circuits: Structural Insights and Techniques

8.1 Structural Results

These are the steps to be undertaken during preprocessing.

Converting to a bivariate polynomial

Proposition 8.1. *A non-commutative polynomial $f \in \mathbb{F}\langle z_1, ..., z_n\rangle$ of degree d having a circuit of size s can be encoded into a bivariate polynomial $f' \in \mathbb{F}\langle x, y\rangle$ of degree almost $(n+2)d$ and size $O(s)$, where x and y are non-commuting variables.*

Proof. Let $f = \sum_{i=1}^{t} c_i w_i$ where $c_i \in$ and each w_i is a monomial in variables $\{z_1, ..., z_n\}$. Encode each z_i using the following substitution: $\forall i \in [n] : z_i \mapsto xy^i x$. Thus each w_i is encoded as a as some $\widetilde{w_i}$ in two variables. Moreover, the substitution map is bijective.
Injectivity: Let $w = z_{i_1} z_{i_2} \dots z_{i_k}$ and $w' = z_{j_1} z_{j_2} \dots z_{j_m}$ be two different monomials. These are mapped to $\widetilde{w} = xy^{i_1} xxy^{i_2} x \dots xy^{i_k} x$ and $\widetilde{w'} = xy^{j_1} xxy^{j_2} x \dots xy^{j_m} x$ respectively. The corresponding exponent sequences $(i_1, i_2, \dots, i_k)$ and $(j_1, j_2, \dots, j_m)$ are different whenever $w \neq w'$, and thus $\widetilde{w} \neq \widetilde{w'}$.

Surjectivity: We are given an arbitrary monomial of the form $xy^{i_1}xxy^{i_2}x\dots xy^{i_k}x$. The sequence $(i_1, i_2, \dots, i_k)$ corresponds exactly to the original monomial $w = z_{i_1}z_{i_2}\dots z_{i_k}$.

□

Due to the bijection, if f is given by a black-box access for evaluation on matrices, we can efficiently create from it a black-box access for f'.

Circuit to ABP

Proposition 8.2. *Let $\mathcal{C}$ be a noncommutative arithmetic circuit of size s computing $f \in \langle x_1, x_2, \dots, x_n\rangle$. Then there is a non-commutative ABP of width and length at most 2^s that computes the same polynomial f.*

Proof. We define a configuration as a subset of gates that have already been dealt with. To simulate the circuit using an ABP, we track which gates have already been dealt with (that is, evaluated) and what values have already been computed.

Definition 8.3 (Configuration)**.** A configuration is a subset of the set of gates G such that, if a gate g is a member of the configuration, then all the input gates of g are also in the configuration.

They can thus be seen as downward closed subsets. Since there are at most s gates, there are at most 2^s possible subsets, and thus the number of possible configurations is also upper bounded by 2^s.

We construct the ABP as follows: Each node of the ABP corresponds to a configuration $S \subseteq G$ of gates that have been evaluated already. The start node (source) corresponds to the configuration $S = \emptyset$. The accepting node (sink) corresponds to the configuration S where the output gate has been evaluated. From a configuration S, you can go to $S' = S \cup \{g\}$ where $g \notin S$ is a gate with all its inputs in S. The edge is labelled as follows:

- If g is an input gate labelled variable x_i, label the edge x_i
- If g is an input gate labelled constant α, label the edge α

- We can move from one subset of evaluated gates to another by evaluating a new gate whose inputs have already been evaluated. We have three cases here.

    1. If we evaluate an input gate (a variable or a constant), simply label the edge with the corresponding variable or constant.

    $$S \xrightarrow{x_i} S \cup \{g\}$$
    $$S \xrightarrow{\alpha} S \cup \{g\}$$

    2. Suppose we evaluate an addition gate: say, $g = g_1 + g_2$

    Suppose $g_1, g_2 \in S$ such that their values are evaluated. We now evaluate g and from from S to $S \cup \{g\}$. Label the edge by 1 because no new variable is introduced.

    $$S \xrightarrow{\alpha} S \cup \{g\}$$

    The ABP will sum over multiple such paths.

    3. Suppose we evaluate a multiplication gate: say, $g = g_1 \times g_2$. Again, we label the edge by 1 as we move from S to S to $S \cup \{g\}$. However, the structure of the ABP has the product order now: it ensures that g_1's contribution occurs before g_2's in any path that computes g. To sum it up, multiplication is thus realized by sequencing paths.

Length and Width: The width of an ABP is the maximum number of nodes in a layer. Each layer corresponds to a set of configurations of a fixed size k, where k is the number of gates that has been evaluated till then. For $0 \leq k \leq s$, atmost $\binom{s}{k}$ configurations are there. Hence,

$$\text{Width} \leq \max_k \binom{s}{k} \leq 2^s$$

The length of an ABP is the number of layers from source to sink. The size of the configuration is increased one gate per step, and in the worst case, we need to evaluate s gates one by one. However. all possible configurations are being accounted for in a sequence, and this corresponds to the longest path through the configuration space. Thus

$$\text{Length} \leq 2^s$$

□

8.2 Proposed Ideas

We consider the problem of rank concentration for a class of non-commutative polynomials, as a step towards black-box PIT. The input is given to us in the form $f = c^\intercal \prod_{j=1}^{l}(u_i x + u_{i+1} y)$, where $f \in \mathbb{F}\langle x, y\rangle$, and the $u_j \in \mathbb{F}^d$ are known vectors. This corresponds to a non-commutative depth 3 circuit model. A natural generalization to non-commutative ABPs replaces these vectors with matrices.

An overview of the strategy

The high level goal is to demonstrate ℓ-rank concentration: that is to show that the coefficient span of all monomials of $\deg_y < \ell$ terms equals the span of all coefficients.

A typical attempt at proving rank concentration involves the following steps, as seen in 7.

1. Decompose the polynomial, usually as a product of simpler components
2. Apply a suitable map to introduce new dependencies among the monomials
3. Show some *local* concentration: this involves analyzing the low support monomials
4. Use inductive (or structural) arguments to show that *local* concentration implies *global* concentration

The idea is to introduce non-commuting auxiliary elements a, b that commute with the original variables x and y . Although they look like generalized monomials, they rest inside a matrix algebra of suitable dimension.

The map suggested can then be a Hadamard product. Define a map ϕ_ℓ.

$$x \mapsto x_a = a \odot x$$
$$y \mapsto y_b = b \odot y$$

This transformation embeds f into the ring $\mathbb{F}\langle a, b\rangle\langle x, y\rangle$.

Instead of using the sparse PIT algorithm as a subroutine, we try to embed the ideas used in the sparse PIT algorithm. We simulate the substitution automaton indirectly.

Given an input monomial w and the current state q_i, reading some symbol $x_b \in \{x, y\}$ the automaton has two choices:

1. **Don't care transition**: This interprets the current symbol as not being a part of the isolating set and emits a block variable ξ_{i+1}.
2. **Step-down Path transition**: This interprets the current symbol as a part of the isolating index set, and emits an index variable $y_{b,i+1}$, and moves to the next state.

This path is thus the one in which the automaton follows a sequence of transitions that include exactly one "step-down" for each $i \in [k]$, identifying $k = \log t$ positions of the monomial as the isolating indices.

We now define the following term.

Definition 8.4 (y- stepdown path)**.** ysp_ℓ is the term that corresponds to picking the stepdown entry in all the y factors.

We hypothesize that the image of f under ϕ_ℓ when analyzed using this idea will exhibit ℓ-rank concentration. In particular, we expect the stepdown path term to act as a certificate as non-zeroness- isolating a non-zero contribution to the coefficient space.

Chapter 9

Conclusion, Open Questions & Future Directions

In this thesis we addressed the question of polynomial identity testing of non-commutative arithmetic circuits. Chapter 3 introduces essential concepts from non-commutative algebra and polynomial identity algebras for a general computer science audience. We also present a self-contained proof of the Amitsur-Levitzki theorem using combinatorial techniques, exploiting the properties of s_n. Chapter 4 presents an important structural property of non-commutative rational functions and proves their invertibility over matrix algebras. We can now see these functions as natural objects living in skew fields. The latter chapters initiate a study of rank concentration for polynomials computed by non-commutative arithmetic circuits.A major challenge during the course of the thesis was to come up with a suitable computational model as well as a corresponding notion of rank concentration. We propose a candidate for the transformation as well as a framework that relies on existing sparse PIT ideas.

Open Questions

- **Rank Concentration Proof:** The central open problem remains to prove that the proposed map induces ℓ-rank concentration
- **Lower Bounds:** Proving the rank concentration conjecture and the black-box PIT algorithm would lead to better lower bounds for the standard polynomial s_n over

suitable matrix algebras. This would then be the first hardness result of this type: the standard polynomial is a *hard* polynomial, much like the permanent polynomial in the commutative setting.

Future Directions

- The study of Polynomial Identity Algebras in the context of nc-PIT remains largely underdeveloped. Beyond the Amitsur-Levitzki Theorem and the result on invertibility, there has been little meaningful interaction between these two areas. I firmly believe that a deeper understanding of PI algebras could yield valuable insight into the nc-PIT problem.

- In Chapter 7, we briefly introduced the notion of Basis Isolating Weight Assignment in the context of (commutative) arithmetic circuits. It remains to see if a similar notion of weight assignment based basis isolation can be observed in the non-commutative setting as well. This method relies on assigning "weights" to monomials, and there seems to be no natural way to separate "weighted" generalized monomials. Developing a meaningful notion of "weight" for generalized monomials may enable a new class of rank concentration maps

Bibliography

[Agr+14] Manindra Agrawal et al. *Hitting-sets for ROABP and Sum of Set-Multilinear circuits*. 2014. arXiv: 1406.7535 [cs.CC]. URL: https://arxiv.org/abs/1406.7535.

[AKS04] Manindra Agrawal, Neeraj Kayal, and Nitin Saxena. "Primes is in P". In: *Annals of Mathematics* 160 (June 2004). Godel Prize, Fulkerson Prize, pp. 781–793. URL: https://www.microsoft.com/en-us/research/publication/primes-is-in-p/.

[AL50] A. S. Amitsur and J. Levitzki. "Minimal Identities for Algebras". In: *Proceedings of the American Mathematical Society* 1.4 (1950), pp. 449–463. ISSN: 00029939, 10886826. URL: http://www.jstor.org/stable/2032312 (visited on 09/22/2024).

[Ami71] S. A. Amitsur. "A note on pi-rings". In: *Israel Journal of Mathematics* 10.2 (June 1971), pp. 210–211. ISSN: 1565-8511. DOI: 10.1007/bf02771571. URL: http://dx.doi.org/10.1007/BF02771571.

[AMS08] Vikraman Arvind, Partha Mukhopadhyay, and Srikanth Srinivasan. "New results on Noncommutative and Commutative Polynomial Identity Testing". In: *CoRR* abs/0801.0514 (2008). arXiv: 0801.0514. URL: http://arxiv.org/abs/0801.0514.

[Art11] M. Artin. *Algebra*. Pearson Education, 2011. ISBN: 9780132413770. URL: https://books.google.co.in/books?id=QsOfPwAACAAJ.

[Art99] M. Artin. *Noncommutative Rings*. https://math.mit.edu/~zyun/Artin_notes.pdf. [Accessed 20-04-2025]. 1999.

[Arv+19] Vikraman Arvind et al. "Randomized Polynomial-Time Identity Testing for Noncommutative Circuits". In: *Theory of Computing* 15.7 (2019), pp. 1–36. DOI: 10.4086/toc.2019.v015a007. URL: https://theoryofcomputing.org/articles/v015a007.

[ASS13] Manindra Agrawal, Chandan Saha, and Nitin Saxena. "Quasi-polynomial hitting-set for set-depth- formulas". In: *Proceedings of the forty-fifth annual ACM symposium on Theory of Computing*. STOC'13. ACM, June 2013, pp. 321–330. DOI: 10.1145/2488608.2488649. URL: http://dx.doi.org/10.1145/2488608.2488649.

[Bre24] Matej Brešar. *The Wedderburn-Artin Theorem*. 2024. arXiv: 2405.04588 [math.RA]. URL: https://arxiv.org/abs/2405.04588.

[BW05] A. Bogdanov and H. Wee. "More on noncommutative polynomial identity testing". In: *20th Annual IEEE Conference on Computational Complexity (CCC'05)*. 2005, pp. 92–99. DOI: 10.1109/CCC.2005.13.

[DF04] Vesselin Drensky and Edward Formanek. *Polynomial Identity Rings*. Birkhäuser Basel, 2004. ISBN: 9783034879347. DOI: 10.1007/978-3-0348-7934-7. URL: http://dx.doi.org/10.1007/978-3-0348-7934-7.

[For72] Edward Formanek. "Central polynomials for matrix rings". In: *Journal of Algebra* 23.1 (Oct. 1972), pp. 129–132. ISSN: 0021-8693. DOI: 10.1016/0021-8693(72)90050-6. URL: http://dx.doi.org/10.1016/0021-8693(72)90050-6.

[Gur+17] Rohit Gurjar et al. "Deterministic Identity Testing for Sum of Read-Once Oblivious Arithmetic Branching Programs". In: *computational complexity* 26.4 (Dec. 2017), pp. 835–880. ISSN: 1420-8954. DOI: 10.1007/s00037-016-0141-z. URL: https://doi.org/10.1007/s00037-016-0141-z.

[Hya77] Laurent Hyafil. "The power of commutativity". In: *18th Annual Symposium on Foundations of Computer Science (sfcs 1977)*. 1977, pp. 171–174. DOI: 10.1109/SFCS.1977.31.

[KI04] Valentine Kabanets and Russell Impagliazzo. "Derandomizing Polynomial Identity Tests Means Proving Circuit Lower Bounds". In: *computational complexity* 13.1 (Dec. 2004), pp. 1–46. ISSN: 1420-8954. DOI: 10.1007/s00037-004-0182-6. URL: https://doi.org/10.1007/s00037-004-0182-6.

[KSS14] Swastik Kopparty, Shubhangi Saraf, and Amir Shpilka. "Equivalence of Polynomial Identity Testing and Deterministic Multivariate Polynomial Factorization". In: *2014 IEEE 29th Conference on Computational Complexity (CCC)*. 2014, pp. 169–180. DOI: 10.1109/CCC.2014.25.

[Lam01] T. Y. Lam. *A First Course in Noncommutative Rings*. Springer New York, 2001. ISBN: 9781441986160. DOI: 10.1007/978-1-4419-8616-0. URL: http://dx.doi.org/10.1007/978-1-4419-8616-0.

[LLS18] Guillaume Lagarde, Nutan Limaye, and Srikanth Srinivasan. "Lower Bounds and PIT for Non-commutative Arithmetic Circuits with Restricted Parse Trees". In: *computational complexity* 28.3 (Sept. 2018), pp. 471–542. ISSN: 1420-8954. DOI: 10.1007/s00037-018-0171-9. URL: http://dx.doi.org/10.1007/s00037-018-0171-9.

[Mor96] Patrick Morandi. *Field and Galois Theory*. Springer New York, 1996. ISBN: 9781461240402. DOI: 10.1007/978-1-4612-4040-2. URL: http://dx.doi.org/10.1007/978-1-4612-4040-2.

[Nis91] Noam Nisan. "Lower bounds for non-commutative computation". In: *Proceedings of the Twenty-Third Annual ACM Symposium on Theory of Computing*. STOC '91. New Orleans, Louisiana, USA: Association for Computing Machinery, 1991, pp. 410–418. ISBN: 0897913973. DOI: 10.1145/103418.103462. URL: https://doi.org/10.1145/103418.103462.

[Pas11] Donald S. Passman. *The algebraic structure of group rings*. Dover Publications, 2011.

[Row74] Louis Halle Rowen. "On rings with central polynomials". In: *Journal of Algebra* 31.3 (Sept. 1974), pp. 393–426. ISSN: 0021-8693. DOI: 10.1016/0021-8693(74)90122-7. URL: http://dx.doi.org/10.1016/0021-8693(74)90122-7.

[RS04] R. Raz and A. Shpilka. "Deterministic polynomial identity testing in non commutative models". In: *Proceedings. 19th IEEE Annual Conference on Computational Complexity, 2004.* 2004, pp. 215–222. DOI: 10.1109/CCC.2004.1313845.

[Sha92] Adi Shamir. "IP = PSPACE". In: *J. ACM* 39.4 (Oct. 1992), pp. 869–877. ISSN: 0004-5411. DOI: 10.1145/146585.146609. URL: https://doi.org/10.1145/146585.146609.