DERANDOMIZING SOME NUMBER-THEORETIC AND ALGEBRAIC ALGORITHMS.

A Thesis Submitted

in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

by

Neeraj Kayal

to the

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY, KANPUR

August, 2006

CERTIFICATE

Certified that the work contained in the thesis entitled "*Derandomizing some number-theoretic and algebraic algorithms.*", by "*Neeraj Kayal*", has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

(Dr. Manindra Agrawal)
Professor,
Department of Computer Sc. & Engg.,
Indian Institute of Technology,
Kanpur.

August, 2006

Synopsis

The aim of this thesis is to provide deterministic upper bounds for various naturally occuring computational problems.

We begin this thesis with a study of computational problems related to rings and their automorphisms. We consider the problem GroupRA of computing the automorhism group of a given ring in terms of a set of generators of its automorphism group. We show that an efficient deterministic algorithm for GroupRA would imply the existence of efficient deterministic algorithms for a number of well-studied problems of intermediate complexity including polynomial factoring (over finite fields), Integer factoring and Graph Isomorphism. On the other hand, we upper bound the complexity of GroupRA by showing that GroupRA is in the complexity class fnAM and therefore is not NP-hard (NP$\not\subseteq$ P$^{\text{GroupRA}}$) unless the polynomial hierarchy collapses. We then consider the problem of computing a nontrivial automorphism of a given ring R and show that it is random-polynomial-time equivalent to integer factoring. We then investigate the complexity of determining the existence of a nontrivial automorphism of a given ring. This problem is shown to admit an efficient deterministic algorithm.

We then study the identity testing problem for depth 3 arithmetic circuits ($\Sigma\Pi\Sigma$ circuits). We give the first deterministic polynomial time identity test for $\Sigma\Pi\Sigma$ circuits with bounded top fanin.

Next, we consider the deterministic complexity of the problem of polynomial factorization over finite fields - given a finite field $\mathbb{F}_q$ and a polynomial $h(x, y) \in \mathbb{F}_q[x, y]$ compute a nontrivial factor of $h(x, y)$. This problem admits a randomized polynomial-time algorithm and no deterministic polynomial-time algorithm is known. We give a deterministic polynomial-time algorithm that *partially* factors the input polynomial $h(x, y)$.

The motivation for the partial factoring algorithm developed is to upper bound the complexity of the following polynomial solvability problem: given a finite field $\mathbb{F}_q$ and a set of polynomials $f_1, f_2, \cdots, f_m \in \mathbb{F}_q[x_1, x_2, \cdots, x_n]$ of total degree at most d determine the $\mathbb{F}_q$-solvability of the system $f_1 = f_2 = \cdots = f_m = 0$. This problem is easily seen to be NP-complete even when the field size q is as small as 2 and the degree of each polynomial is bounded by $d = 2$. Here we investigate the deterministic complexity of this problem when the number of variables n in the input is bounded. We show that there is a *deterministic* algorithm for this problem whose running time, for any fixed n, is bounded by a polynomial in d, m and $\log q$. Moreover, the algorithm can be implemented parallely to get a family of P -uniform circuits of depth $\text{poly}(\log d \cdot \log m \cdot \log q)$ and size $\text{poly}(d \cdot m \cdot \log q)$ for the solvability problem.

Finally, we present a deterministic polynomial-time algorithm that determines whether an input number n is prime or composite.

Acknowledgements

Over the years, many teachers have initiated and guided me into the world of science. I will try to acknowledge some of them. Naturally, my warmest thanks are to my advisor Manindra Agrawal.

There is so much that I want to thank him for that I am afraid I will not be able to do a good enough job. I am sure that in the silent understanding between us, Manindra is more aware of the extent of friendship, admiration and gratitude I have for him than I am able to express in words. But this dissertation will hopefully be read my many people other than Manindra so let me try.

Thank you Manindra for your close guidance in all different aspects of the scientific process. In all things, large and small, I knew that I could always count on you!! Thank you for having had confidence in me from the first time I stepped into your office as an undergraduate student. Thank you for treating me as a colleague from the first time we started working on a common problem and at the same time sheltering me in a parental manner, especially during a time of illness. Thank you for pushing me forward when possible and laying off in times work could not have been a high priority for me. And thank you for sharing with me in hours and hours of conversations over coffee, your understanding of computer science, your zest for clarity and simplicity of exposition and an abundance of ideas.

I am grateful to IIT Kanpur, especially the Department of Computer Science and Engineering for providing a wonderful environment where learning was possible. Specially Professor Somenath Biswas, Sumit Ganguly and Pankaj Jalote for their teaching and encouragement during the years.

During the high-school days, many teachers sparked in me an intrest in science, teachers whose guidance and mentoring was priceless. For this, I am especially indebted to Mangesh Rege and to Jagadis Bose National Science Talent Search (JBNSTS).

My colleague and friend Nitin made the entire research process very enjoyable.I learnt quite a bit of my mathematics and computer science from him.

I am grateful to Bernard Chazelle and Princeton University for hosting me in 2003-04. I am also thankful to P. S. Thiagarajan and National University of Singapore for hosting me in 2004-05. There are so many other people to whom I am in debt for sharing their knowledge and ideas with me. A partial list that comes to my mind is: Hendrik Lenstra, Jaikumar Radhakrishnan, K V Subramaniam, Alan Lauder and Amit Sahai.

A special thanks to my office-mate, friend and counsellor Atul Gupta for all the wonderful time I have spent with him. I have also been fortunate to have had a large number of friends despite moving from one university to another at the end of each academic year. I will thank them in person.

Finally, I have to say thank you to my family, particularly my father and mother for everything they have done for me. They raised me, supported me, taught me, and loved me. To them I dedicate this thesis.

Contents

Chapter 1

Introduction

The aim of this work is to give *deterministic upper bounds* for some naturally occuring computational problems. The computational problems that we consider typically have a strong algebraic and/or number-theoretic flavour. In order to understand the task ahead of us - its significance and its limitations - let us first consider the role of *randomness* in computation. A randomized algorithm is an algorithm that is allowed to "flip coins". Thus at each step of its computation, a randomized algorithm can obtain a bit which is 0 with probability half and 1 with probability half (and is independent of previous coin-flips). For some tasks such as those of achieving cryptographic security or simulating probabilistic events, randomness is essential and cannot be dispensed with. For algorithmic tasks it is generally believed that *randomness can be dispensed with, without significantly increasing the usage of other resources such as time, space or parallelism.* This belief, however remains an unproven conjecture. The task of proving concrete versions of this conjecture is referred to as *derandomization*. Research on derandomization has proceeded along two lines. One stream of research makes use of complexity-theoretic assumptions to derandomize entire complexity classes. An example is the result by Impagliazzo and Wigderson [IW97] showing that if there exists a language in the complexity class E requiring exponential size circuits then P = BPP. The other stream of derandomization research seeks to establish deterministic upper bounds *for specific computational problems* without relying on any unproven assumptions. An example is the recent result by Omer Reingold [Rei05] showing that undirected connectivity is in LOGSPACE. The work that we present in this dissertation is of the second kind. In the rest of this chapter we expand on this discussion and then present the specific computational problems that we deal with and give an overview of our results.

1.1 Randomized Algorithms.

Let L $\subseteq \{0,1\}^*$ be a language. An algorithm $\mathcal{A}$ which takes as input a pair of strings $\langle x, r\rangle$ and accepts with *high probability* (over the random choice of r) if and only if $x \in$ L is called a randomized algorithm for the language L. Ever since its invention in the 1970s randomization has played a central role in algorithm design. The class BPP of problems admitting efficient randomized algorithms has now replaced the class P of problems considered efficiently solvable. For most problems, the available randomized algorithms significantly outperform their deterministic counterparts in terms of resource usage (time, space, parallelism) and their ease of implementation.

1.2 Derandomization.

As remarked above, randomized algorithms are usually superior to their deterministic counterparts in most ways. Indeed for each of the problems that we consider here, we do not expect our deterministic algorithms to be practically competitive with the existing randomized ones. It is therefore natural to question the motivation behind the study of derandomization and its importance, if any.

The first justification is common to all of theory. The task of devising deterministic algorithms often poses very elegant and challenging mathematical problems of a fundamental nature. When we design an n^{15} algorithm for a problem, we do not give a practical way to solve that problem. Instead, we are proving something inherent about this problem, namely that it is in P , and so its hardness does not grow exponentially with the input size. The hope is that along the way we will also produce some key insights that might later be used in practical algorithms for it, in improving the analysis of existing algorithms for it or in solutions to other related problems.

The second justification has to do with the nature of the physical world we live in. A randomized algorithm requires a source of truly random bits for its analysis to hold good. Moreover, in practical situations we need to provide the algorithm with these random bits at a huge rate - the speed at which modern computers operate. It is not clear if there does exist a source of true randomness and even if it does, we cannot in practice generate these random bits at the same rate at which they are consumed by the algorithm.

Most important however is the justification that stems from recent discoveries showing that derandomizing specific computational problems would lead to lower bounds. It was shown by Impagliazzo and Kabanets [IK03] that derandomizing a specific BPP-problem, Polynomial Identity Testing, is essentially *equivalent* to proving arithmetic circuit lower bounds for NEXP. More recently, it was shown in [Agr05] that a "black-box derandomization" of the identity testing problem would imply strong arithmetic circuit lower bounds. For a precise definition of black-box derandomization of a randomized algorithm see the chapter on primality testing, chapter 7. Proving absolute lower bounds on the complexity of problems is the central aim of complexity theory. This connection between proving lower bounds and devising deterministic algorithms provides the main motivation for the latter. In general, the hope is that devising deterministic algorithms for specific computational problems would eventually lead to the development of tools and techniques that are useful in proving lower bounds.

1.3 The Problems.

The problems that we *attempt* to derandomize are the following:

- **Polynomial Factoring over finite fields.** Given a finite field $\mathbb{F}_q$ and a polynomial $f(x_1, \ldots, x_n) \in \mathbb{F}_q[x_1, \ldots, x_n]$, find a nontrivial factor of f.
- **Solvability.** Given a finite field $\mathbb{F}_q$ and a set of polynomials

  $$f_1(x_1, \ldots, x_n), \ldots, f_m(x_1, \ldots, x_n) \in \mathbb{F}_q[x_1, \ldots, x_n],$$

  determine if there is a common $\mathbb{F}_q$-solution to the system of equations

  $$f_1(\bar{\mathbf{x}}) = \ldots = f_m(\bar{\mathbf{x}}) = 0.$$

- **Identity Testing.** Given a field $\mathbb{F}$ and an arithmetic circuit $\mathcal{C}$ over $\mathbb{F}$, determine if the polynomial computed by it is the identically zero polynomial.
- **Primality Testing.** Given a number n, determine whether it is prime or not.

We will also be considering the following problems.

- **Integer Factoring.** Given an integer n, compute its prime factorization.

- **Graph Isomorphism.** Given two graphs G_1 and G_2, determine if they are isomorphic.
- **Ring Automorphism.** Given a finite ring R as input, determine if it admits a nontrivial automorphism.
- **GroupRA.** Given a finite ring R as input, compute its automorphism group $Aut(R)$ in terms of a set of generators for $Aut(R)$.

1.4 Our Contributions

We devise deterministic algorithms for some specific computational problems which were previously known to admit only randomized algorithms. The computational resources that we will be concerned with are time and parallelism. We devise a deterministic algorithm for *partially* factoring polynomials. Our algorithm is an extension of deterministic distinct degree factoring algorithms that were available earlier [GKL04]. Moreover, it is efficiently parallelizable with respect to the degree of the input polynomial. We then use it to devise an efficient deterministic algorithm for Solvability *when the number of variables in the input is bounded.* Moreover the parallel time complexity of the algorithm is polylogarithmic in the degree of the input polynomials and the number of equations. We then consider the problem of identity testing and devise an efficient deterministic algorithm for depth-3 arithmetic circuits with bounded top fanin. Finally, we present a "black-box derandomization" of the randomized primality testing algorithm of Agrawal and Biswas [AB03].

A common mathematical object that recurs in all these algorithms is a ring and the group of automorphisms associated with it. So we begin our study with an investigation of the complexity of computing the automorphism group of a given ring and some variants of this problem. We show that many important and well-studied computational problems of "intermediate complexity" such as polynomial factoring, integer factoring and graph isomorphism *deterministically* reduce to the problem of computing the automorphism group of a given ring. We also show that computing a single nontrivial automorphism is (random polynomial time) equivalent to integer factoring. Finally we show that *determining the existence of a nontrivial automorphism* can be done in deterministic polynomial time.

Remark. ZPP $\subseteq$ BPP is the class of languages admitting randomized algorithms whose output is always correct and whose *expected running time* is polynomial in the input size.

Primality Testing and Solvability (in a bounded number of variables) over finite fields were the only two "natural decision problems" known to be in the complexity class ZPP but not known to be in P . We give efficient deterministic algorithms for both these problems and now, we do not know the example of any such problem in ZPP not known to be in P .

Remark. We do not expect our deterministic algorithms to be practically competitive with the existing randomized algorithms. Consequently, we shall often forego some optimizations that would have been practically very significant, were our deterministic algorithms to be implemented. For a given computational problem at hand, we shall instead strive to present a simple and *efficient* deterministic algorithm with a proof of correctness that is elementary, short and self-contained. We will also not be doing a detailed time-complexity analysis of an algorithm being presented as long as it clear that it is "qualitatively efficient". Thus for polynomial-time bounded computation we do not calculate the exact exponent as long as it is clear that the exponent occuring is a constant.

1.5 Organization.

Some of the results presented in this monograph first appeared in the following papers: [AKS04, KS05, Kay05, KS06]. We begin the dissertation with a brief introduction to relevant concepts and theorems from algebra and complexity theory in chapter 2. The subsequent chapters each deal with a specific computational problem and can be followed independently of each other. Finally we collect some open problems and conjectures in chapter 8.

Chapter 2

Preliminaries

Summary:

In this chapter, we give a brief introduction to relevant concepts and theorems from algebra and complexity theory.

2.1 Algebraic Preliminaries.

In this section we mention some elementary properties of groups and rings. For further details, the interested reader is referred to the text by Herstein [Her75] and McDonald [McD74].

2.1.1 Groups

A group $(G, \cdot)$ consists of a set of elements G together with a multiplication operation '$\cdot$' satisfying the properties:

- **Associative Law.** $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ for all $a, b, c \in G$.
- **Existence of identity.** There exists an *identity element* e such that $a \cdot e = a$ for all $a \in G$.
- **Existence of inverse.** Every element a has an *inverse* which is an element b such that $a \cdot b = 1$.

A group $\langle G, \cdot \rangle$ is said to be *commutative* if $a \cdot b = b \cdot a$ for all $a, b \in G$. The group operation for a commutative group is often denoted by '$+$' rather than a '$\cdot$'. If G, H are two groups then we use $H \leq G$ to denote that H is a subgroup of G. For a finite group G: $H \leq G$

implies that $\#H$ divides $\#G$ (Lagrange's Theorem). Given two groups $\langle G_1, \cdot\rangle$ and $\langle G_2, \cdot\rangle$ we can compose them to get a new group $G_1 \oplus G_2$ whose set of elements is $G_1 \times G_2$ and the group operation '$\cdot$' is done component-wise. That is,

$$(g_1, g_2) \cdot (g_1', g_2') = (g_1 \cdot g_1', g_2 \cdot g_2').$$

▮ *Sylow subgroups.*

If d is a divisor of the size of some finite group G then it is not true in general that G has a subgroup of size d. But for a prime p, if $p^k|\#G$ then there always exist a subgroup of size p^k. If p^k is the highest power of p dividing $\#G$ then a subgroup of size p^k is called a *p-Sylow subgroup* of G. In general a group G does not have a unique p-Sylow subgroup, but it does indeed hold true that all the p-Sylow subgroups of G are conjugates of each other (cf. Herstein [Her75]). A p-Sylow subgroup S_p of size p^k can be broken into a *composition series*, i.e., there are groups G_i of size p^{k-i} such that:

$$S_p = G_0 \ > \ G_1 \ > \ G_2 > \ldots \ > \ G_k = \{1\}.$$

▮ *Structure Theorem for finite commutative groups.*

There is a classification known for finite commutative groups. Basically, each such group completely decomposes into a bunch of *cyclic* groups.

Proposition 2.1.1. [Structure theorem for finite commutative groups] *If $\langle G, +\rangle$ is a finite commutative group then it can be uniquely (up to permutations) expressed as:*

$$(G, +) \cong \bigoplus_i (\mathbb{Z}/{p_i}^{\alpha_i}\mathbb{Z})$$

where p_i's are primes (not necessarily distinct) and $\alpha_i \in \mathbb{Z}^{\geq 1}$ and $\oplus$ is the natural composition of groups with component-wise multiplication.

2.1.2 Rings

A *ring* $(R, +, .)$ consists of a set of elements together with addition and multiplication operations. The basic properties of a ring are

- $(R, +)$ is an commutative group.

- **Multiplicative Identity.** There exists an element $1 \in R$ such that $1 \cdot r = r$ for all $r \in R$.
- **Associative law.** $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ for all $a, b, c \in R$.
- **Distributive law.** $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$ and $(b + c) \cdot a = (b \cdot a) + (c \cdot a)$ for all $a, b, c \in R$.

Throughout this thesis, the rings that we come across will always have the (multiplicative) identity element. A ring is said to be *commutative* if $a \cdot b = b \cdot a$ for all elements $a, b \in R$. All the rings that we come across in this dissertation will be commutative unless mentioned otherwise. There are two useful groups living in a ring R. Firstly, $(R, +)$ is a group with respect to addition called the *additive group.* If R^* is the set of elements in R having multiplicative inverse then $(R^*, \cdot)$ is the second group called the *multiplicative group.* For precise definition and some elementary properties see Herstein [Her75].

In analysing a ring R we use special subgroups of $(R, +)$ called *ideals.*

Definition 2.1.2. A subset $I \subseteq R$ is an *ideal* of R if:

- $(I, +)$ is a subgroup of $(R, +)$, and
- for all $i \in I,\ r \in R$, both $i \cdot r$ and $r \cdot i$ are in I. This can also be stated as: $\forall r \in R$ both $r \cdot I,\ I \cdot r \subseteq I$.

Ideals can be multiplied together to give new (smaller) ideals.

Definition 2.1.3. Let $\mathcal{I}, \mathcal{J}$ be two ideals of a ring R. We define their product as:

$$\mathcal{I} \cdot \mathcal{J} := \text{ ring generated by the elements } \{ij \mid i \in \mathcal{I}, j \in \mathcal{J}\}$$

Powering of ideals, $\mathcal{I}^t$ for positive integer t, is defined similarly. It is easy to see that $\mathcal{I} \cdot \mathcal{J}$ is again an ideal of R.

2.1.3 Local Rings

We will often come across special kinds of rings known as *local* rings. The importance of local rings stems from the structure theorem for finite commutative rings which states that any finite commutative ring can be written uniquely as a direct sum of local rings. We now define local rings and mention their elementary properties. We refer the interested reader to [McD74] for further properties of such rings.

Definition 2.1.4. A commutative ring R is said to be a *local ring* if it has a unique maximal ideal.

Example: Consider ring $R = \mathbb{F}[x_1, x_2]/(x_1^3, x_2(x_2 + x_1))$. Observe that R is a local ring with the unique maximal ideal $\mathcal{M}$ generated by x_1, x_2. Also note that $\mathcal{M}$ is the set of *nilpotent* elements, i.e., for any element $m \in \mathcal{M}$ there is a $k \geq 1$ such that $m^k = 0$ in R.

The basic properties of finite commutative local rings are captured in the following lemma.

Lemma 2.1.5. *([McD74].) Let R be a finite commutative local ring whose additive group $(R, +)$ is the direct sum of d cyclic groups. Then:*

(i). Every non-unit element $r \in R$ is a nilpotent *element, i.e. $r^t = 0$ for some $t \in \mathbb{Z}_{\geq 1}$.*

(ii). The unique maximal ideal $\mathcal{M}$ of R consists of all the nilpotent elements of R.

(iii). There is an integer $t \leq d$ such that the product of any t (not necessarily distinct) elements of $\mathcal{M}$ is zero in R.

(iv). If R is an $\mathbb{F}$-algebra for some finite field $\mathbb{F}$, then every element $r \in R$ can be uniquely written as $r = \alpha + m$, $\alpha \in \mathbb{F}$ and $m \in \mathcal{M}$. This implies that there is a unique ring homomorphism $\phi : R \to \mathbb{F}$ such that $\phi(\alpha + m) = \alpha$.

There is also a converse to part (i) of the above lemma. For a finite commutative ring R, if every non-unit element $r \in R$ is a *nilpotent* element then R is a local ring.

2.1.4 Structure Theorem for finite commutative rings.

Algebraic structures often *break* into simpler objects. In case of rings, these *elementary structures* are referred to as *indecomposable* rings.

Definition 2.1.6. Indecomposable ring: A ring R is said to be indecomposable if there do not exist rings R_1, R_2 such that $R \cong R_1 \otimes R_2$, where $\otimes$ denotes the natural composition of two rings with component wise addition and multiplication.

Remark. Here and henceforth we will use '$\oplus$' to denote group composition and '$\otimes$' to denote ring composition. In mathematical literature '$\otimes$' is sometimes used to denote the tensor-product of rings. In this dissertation, we will never come across tensor-products and we will always use '$\otimes$' to denote the natural composition of two rings with component-wise addition and multiplication.

Theorem 2.1.7. *A finite commutative ring is indecomposable if and only if it is local.*

Unlike commutative groups, a classification of commutative rings is not known yet. But as a first step rings can be decomposed uniquely into indecomposable rings.

Proposition 2.1.8. [Structure theorem for rings] [McD74]. *If R is a finite ring then it uniquely (up to permutations) decomposes into indecomposable rings $R_1, \dots, R_s$ such that*

$$R \cong R_1 \otimes \ldots \otimes R_s$$

2.1.5 Examples.

Example Let $n = p^2q$ where p, q are distinct primes and define a natural ring $R := (\mathbb{Z}/n\mathbb{Z}, +, \cdot)$. Then observe that R decomposes as $(\mathbb{Z}/p^2\mathbb{Z}, +, \cdot) \otimes (\mathbb{Z}/q\mathbb{Z}, +, \cdot)$ where the two *component* rings are local. ∎

Example Consider a ring $R := \mathbb{F}[x, y]/(x^3, y^2)$. The subset yR, denoted as (y), is an ideal of R. Similarly, $xR + yR$, denoted by (x, y), is also an ideal of R. Note that the product of these two ideals is $(y) \cdot (x, y) = (xy, y^2) = (xy)$. Similarly in R, $(x, y)^2 = (x^2, xy)$, $(x, y)^3 = (x^2y)$ and $(x, y)^4 = 0$. Moreover, it can be shown that R is a local ring with $\mathcal{M} = (x, y)$ as its unique maximal ideal. ∎

Example Consider the ring $R := (\mathbb{Z}/p^2q^3\mathbb{Z})[x, y]/(x^4, px, y^2 - y)$. By factoring the characteristic p^2q^3 we get:

$$R \cong (\mathbb{Z}/p^2\mathbb{Z})[x, y]/(x^4, px, y^2 - y) \otimes (\mathbb{Z}/q^3\mathbb{Z})[x, y]/(x^4, px, y^2 - y)$$

Further, by factoring $y^2 - y$ into *coprime* irreducibles over the respective local rings in x we get:

$$\begin{aligned} R \;\cong\; & (\mathbb{Z}/p^2\mathbb{Z})[x, y]/(x^4, px, y) \otimes (\mathbb{Z}/p^2\mathbb{Z})[x, y]/(x^4, px, y - 1) \\ & \otimes (\mathbb{Z}/q^3\mathbb{Z})[x, y]/(x^4, px, y) \otimes (\mathbb{Z}/q^3\mathbb{Z})[x, y]/(x^4, px, y - 1) \end{aligned}$$

∎

2.1.6 Representing Rings

For concreteness we first fix the way we are going to present the finite rings and their homomorphisms in the input or the output.

Definition 2.1.9. Basis representation of rings: A finite ring R is given by first describing its additive group in terms of n additive generators and then specifying multiplication by giving for each pair of generators, their product as an element of the additive group. More concretely, R is presented as:

$$(R,+,.) := \langle (d_1, d_2, d_3, \cdots, d_n), ((a_{i,j,k}))_{1 \leq i,j,k \leq n} \rangle$$

where, for all $1 \leq i,j,k \leq n,\ 0 \leq a_{i,j,k} < d_k$ and $a_{i,j,k} \in \mathbb{Z}$.

This specifies a ring R generated by n elements $b_1, b_2, \cdots b_n$ with each b_i having additive order d_i and $(R,+) = (\mathbb{Z}/d_1\mathbb{Z})b_1 \oplus (\mathbb{Z}/d_2\mathbb{Z})b_2 \cdots \oplus (\mathbb{Z}/d_n\mathbb{Z})b_n$. Moreover, multiplication in R is defined by specifying the product of each pair of additive generators as an integer linear combination of the generators: for $1 \leq i,j \leq n$, $b_i \cdot b_j = \sum_{k=1}^{n} a_{i,j,k} b_k$.

Definition 2.1.10. Representation of maps on rings: Suppose R_1 is a ring given in terms of its additive generators $b_1, \ldots, b_n$ and ring R_2 given in terms of $c_1, \ldots, c_n$. In this chapter maps on rings would invariably be homomorphisms on the additive group. Then to specify any map $\phi : R_1 \rightarrow R_2$, it is enough to give the images $\phi(b_1), \ldots, \phi(b_n)$. So we represent ϕ by an $n \times n$ matrix of integers A, such that for all $1 \leq i \leq n$:

$$\phi(b_i) = \sum_{j=1}^{n} A_{i,j} c_j$$

and for all $1 \leq i,j \leq n,\ 0 \leq A_{i,j} <$ additive order of c_j.

Example Consider the ring $R := (\mathbb{Z}/3\mathbb{Z})[x]/(x^2 - x + 1)$. Here, 1 and x can be taken as basis elements and $(R,+) = (\mathbb{Z}/3\mathbb{Z}) \cdot 1 \oplus (\mathbb{Z}/3\mathbb{Z}) \cdot x$. Multiplication on the basis elements is defined as: $1 \cdot 1 = 1 \cdot 1 + 0 \cdot x,\ 1 \cdot x = x \cdot 1 = 0 \cdot 1 + 1 \cdot x$ and $x \cdot x = 2 \cdot 1 + 1 \cdot x$. Note that the map ϕ sending $1 \mapsto 1$ and $x \mapsto -1$ is a homomorphism from R to itself and with respect to the basis $\{1, x\}$ it can be represented as: $A = \begin{pmatrix} 1 & 0 \\ 2 & 0 \end{pmatrix}$. ∎

2.1.7 Hensel Lifting Lemma

We collect here the well-known Hensel-lifting lemma which is used at several places in this monograph. Let R be a ring and $f \in R$ be an element of R. Further let $\mathcal{I}$ be an ideal of R. The Hensel lifting lemma then says that given a '*coprime*' factorization of f in the ring $R/\mathcal{I}$, that factorization can be uniquely '*lifted*' to a factorization in the ring $R/(\mathcal{I}^2)$.

To make the above statement more precise we need to extend the definition of coprimality for arbitary rings.

Definition 2.1.11. Let R be a ring and $a, b \in R$ be two elements in R. Then a and b are said to be *formally coprime* if there exist $a', b' \in R$ such that $a \cdot a' + b \cdot b' = 1$.

Lemma 2.1.12. [1] *Let R be a ring and $\mathcal{I} \subset R$ be an ideal. For any $f \in R$ and for any factorization $f = g \cdot h \ (mod\, \mathcal{I})$ of f in $R/\mathcal{I}$ such that $g, h \in R/(\mathcal{I})$ are formally coprime in $R/\mathcal{I}$, there exist g^* and h^* in $R/(\mathcal{I}^2)$ such that*

$$\left.\begin{aligned} f &\equiv g^* \cdot h^* \ (mod\, \mathcal{I}^2) \\ g^* &\equiv g \ (mod\, \mathcal{I}) \\ h^* &\equiv h \ (mod\, \mathcal{I}). \end{aligned}\right\} \qquad (1)$$

Moreover, the following holds:

- *For any solution g^*, h^* to (1), g^* and h^* are formally coprime in $R/(\mathcal{I}^2)$.*
- *Given $f \in R$ and $g, h \in R/\mathcal{I}$, we can compute $g^*, h^* \in R/(\mathcal{I}^2)$ by solving a system of linear equations over $R/\mathcal{I}$.*
- *The solution g^*, h^* to (1) is unique in the following sense. Any other pair $g', h' \in R/(\mathcal{I}^2)$ is a solution to (1) if and only if there exists an element $\alpha \in \mathcal{I}$ such that*

$$\begin{aligned} g' &= (1 + \alpha) \cdot g^* \ (mod\, \mathcal{I}^2) \\ h' &= (1 - \alpha) \cdot h^* \ (mod\, \mathcal{I}^2). \end{aligned}$$

2.2 Basics of Complexity Theory

A decision problem in computer science is represented by a *language* $L \subseteq \{0, 1\}^*$ which is the set of all 'yes' strings. We say that L is in the *complexity class* NP if there is a

[1] *This version of the Hensel lemma is taken from the lecture notes on Algebra and Computation by Madhu Sudan available at* `http://theory.lcs.mit.edu/%7Emadhu/FT98/course.html`

polynomial time deterministic Turing Machine M and a positive number c such that:

$$L = \Big\{ x \mid \ \exists y \in \{0,1\}^{|x|^c}, \ M(x,y) \text{ accepts} \Big\}$$

x is the *input* and y is called as *witness*, *membership proof* or *nondeterministic guess*. L is said to be in coNP iff $\overline{L} \in$ NP.

Example Consider the problem of satisfiability of boolean formulas:

$$\text{3-SAT} := \{\phi(x_1, \ldots, x_n) \mid \phi = \wedge_{i=1}^{m} (x_{i_1} \vee x_{i_2} \vee x_{i_3}) \text{ and has a satisfying assignment}\}$$

3-SAT is in NP as given a formula ϕ and a satisfying assignment $\overline{v}$ it can be verified in polynomial time whether $\phi(\overline{v})$ is 'true'. ∎

We can also define a "randomized" version of the class NP called AM (for Arthur-Merlin protocol). We will say a language L is in AM if there is a positive number c and a polynomial time deterministic Turing Machine M such that:

$$\begin{aligned} x \in L \quad &\Rightarrow \quad \text{Prob}_{y \in \{0,1\}^{|x|^c}}[\exists z \in \{0,1\}^{|x|^c}, \ M(x,y,z) \text{ accepts}] \geq \frac{2}{3} \\ x \notin L \quad &\Rightarrow \quad \text{Prob}_{y \in \{0,1\}^{|x|^c}}[\exists z \in \{0,1\}^{|x|^c}, \ M(x,y,z) \text{ accepts}] \leq \frac{1}{3} \end{aligned}$$

Typically, the proof of showing an $L \in$ AM goes through by giving a protocol between the *Verifier* (named Arthur – the 'king') who can do randomized polynomial time computations and the *Prover* (named Merlin – the 'advisor' to the king) who has unlimited computational resources. Arthur is interested in determining whether the input $x \in L$ and he sends (x, y) to Merlin who responds with a witness z. Arthur does some computations on (x, y, z) following M and decides whether $x \in L$ with high confidence.

A classic example of a problem in AM is that of checking whether a set is large. We will be referring to its AM protocol in a later chapter.

Proposition 2.2.1. *Suppose S is a set whose membership can be tested in nondeterministic polynomial time and its size is either m or $2m$. Then the decision problem of testing whether S is of size $2m$ is in AM.*

Proof. The idea of the AM protocol is that if S is large then for a random hash function h there will be an $x \in S$ such that $h(x) = 0$ with high probability.

Suppose that the elements of S are represented as binary strings of length s. Arthur first increases the 'gap' in the size of S by defining a new set $T = S^4$. Now $\#T$ is either

m^4 or $16m^4$. Also, the elements of T are binary strings of length $4s$. View them as a column vector. Arthur then chooses a random 0/1 matrix A of size $\lceil \log 3m^4 \rceil \times 4s$ and sends it to Merlin. Merlin returns a column vector $t \in \{0,1\}^{4s}$. Arthur accepts iff $t \in T$ and $A \cdot t = 0 \pmod 2$.

To analyse this AM protocol note that for a given $x \in \{0,1\}^{4s} \setminus \{0\}^{4s}$:

$$\text{Prob}_{A \in \{0,1\}^{\lceil \log 3m^4 \rceil \times 4s}} [A \cdot x = 0 \pmod 2] = \frac{1}{2^{\lceil \log 3m^4 \rceil}}$$

Thus by linearity of expectation:

$$E_{A \in \{0,1\}^{\lceil \log 3m^4 \rceil \times 4s}} [\#\{t \in T \mid A \cdot t = 0 \pmod 2\}] \;=\; \frac{\#T}{2^{\lceil \log 3m^4 \rceil}}.$$

Now Markov inequalities give us that:

$$\begin{aligned} \#T = 16m^4 \quad &\Rightarrow \quad \text{Prob}_{A \in \{0,1\}^{\lceil \log 3m^4 \rceil \times 4s}} [\exists t \in T,\ A \cdot t = 0 \pmod 2] \geq \frac{5}{8} \\ \#T = m^4 \quad &\Rightarrow \quad \text{Prob}_{A \in \{0,1\}^{\lceil \log 3m^4 \rceil \times 4s}} [\exists t \in T,\ A \cdot t = 0 \pmod 2] \leq \frac{1}{3} \end{aligned}$$

This shows that with high probability Arthur accepts only when set S is large.

Also, note that this AM protocol uses $O(s \log m)$ random bits (for A) and $O(s)$ nondeterministic bits (for t). □

If a problem L is in NP $\cap$ coNP then intuition suggests that it should not be "hard". Similarly, if a problem L is in NP$\cap$coAM (or AM $\cap$coAM) then L is 'unlikely' to be NP-hard. What makes these classes interesting is that there are many problems in NP$\cap$coAM that are not known to be in P . Such problems are called problems of "intermediate" complexity. To make these notions more precise we need to form a polynomial-time hierarchy.

Let us denote NP by Σ_1 and define $\Sigma_2 = \text{NP}^{\text{NP}}$, where by $\text{NP}^{\mathcal{C}}$ we mean set of languages L such that there is a polynomial time deterministic Turing Machine M using an *oracle* to $\mathcal{C}$ and a positive number c such that:

$$L = \left\{ x \mid \ \exists y \in \{0,1\}^{|x|^c},\ M(x,y) \text{ accepts} \right\}$$

Similarly, $\Sigma_k := \text{NP}^{\Sigma_{k-1}}$. The union of all these Σ's is called the *polynomial-time hierarchy*: PH $= \cup_{k \geq 1} \Sigma_k$.

It is mostly believed that $\Sigma_1, \Sigma_2, \ldots$ are all distinct complexity classes and hence there is no k such that PH collapses to Σ_k. Coming back to the intermediate complexity classes, it is easy to see that if NP$\cap$coNP has a NP-hard problem then PH $= \Sigma_1$. Also, if NP$\cap$coAM (or AM $\cap$coAM) has a NP-hard problem then it was shown in [Sch88, Kla89] that PH collapses to the second level Σ_2. The proof goes through by showing that AM $\cap$ coAM is *low for* Σ_2, i.e., $\Sigma_2^{\mathrm{AM}\ \cap\mathrm{coAM}} = \Sigma_2$ and thus, NP $\subseteq$ AM $\cap$ coAM implies $\Sigma_3 = \Sigma_2^{\mathrm{NP}} \subseteq \Sigma_2^{\mathrm{AM}\ \cap\mathrm{coAM}} = \Sigma_2$ which eventually results in collapsing PH to Σ_2.

This notion of intermediate complexity can be generalized to *functional problems.* We define FP to be the set of functional problems computable in polynomial time. Define *functional NP* – denoted by fnNP – to contain functions $f : \{0,1\}^* \rightarrow \{0,1\}^*$ such that there is a a positive number c and a deterministic polynomial time Turing machine M (that *outputs* a string) such that, for all $x,\ t \in \{0,1\}^*$:

$$f(x) = t \quad \text{iff} \quad \exists z \in \{0,1\}^{|x|^c}\ M(x,z) = t \tag{2.1}$$

Remark: The above definition says that for all $x,\ t \in \{0,1\}^*$, t is a correct output of the function f on the input x if and only if there is an easily verifiable certificate z certifying that $f(x) = t$. ∎

Now if a function $f \in$ fnNP is NP-hard (SAT $\in$ P ^{f}) then we also get that coSAT $\in fnNP$ implying that unsatisfiability of boolean formulae has short, easily verifiable proofs. That is, coNP $\subseteq$ NP and so the polynomial hierarchy collapses to Σ_1. Thus it is "unlikely" that any function in fnNP is NP-hard.

These notions for functional version of NP can be extended to the class AM which is a randomized version of the class NP. Define *functional AM* – denoted by fnAM – to contain functions $f : \{0,1\}^* \rightarrow \{0,1\}^*$ such that there is a a positive number c and a deterministic polynomial time Turing machine M (that *outputs* a string) such that, for all $x,\ t \in \{0,1\}^*$:

$$f(x) = t \quad \text{iff} \quad \mathrm{Prob}_{y\in\{0,1\}^{|x|^c}}[\exists z \in \{0,1\}^{|x|^c}\ M(x,y,z) = t] \geq \frac{2}{3} \tag{2.2}$$

Remark: The above Definition says that for "most" of the y 's there is a z such that $M(x,y,z)$ outputs the correct value of $f(x)$. On the other hand, for "most" of the y 's there is no z such that $M(x,y,z)$ outputs an incorrect value. ∎

Again the techniques of Schoning [Sch88] essentially show that fnAM is low for Σ_2.

Lemma 2.2.2. (Schoning, [Sch88]) *If* $f : \{0,1\}^* \rightarrow \{0,1\}^* \in fnAM$ *then* $\Sigma_2^f = \Sigma_2$.

Thus, if a function $f \in$ fnAM is NP-hard (i.e. NP $\subseteq$ P ^{f}) then PH collapses to Σ_2. We sketch the proof here for the sake of completeness. Define for all $k \geq 1$, $\Pi_k :=$ co-Σ_k.

Proposition 2.2.3. Schoning, [Sch88]. $\Sigma_2^{fnAM} = \Sigma_2$.

A proof of this proposition is also given in Saxena [Sax06].

2.2.1 Reductions

Some results in this thesis *reduce* one problem L to another problem L'. If there is a function $f : \{0,1\}^* \to \{0,1\}^*$ in class $\mathcal{C}$ such that $x \in L$ iff $f(x) \in L'$ then we say that L is *many-one reducible* to L' and denote it by $L \leq_m^{\mathcal{C}} L'$.

If a problem L can be solved in class $\mathcal{C}$ by using L' as an oracle then we say that L is *Turing reducible* to L' and denote it by $L \leq_T^{\mathcal{C}} L'$.

In the reductions given in this chapter $\mathcal{C}$ is either P or ZPP – the set of languages (functions) that can be decided (computed) in *expected* polynomial time.

Chapter 3

Automorphisms of Rings

Summary: In this chapter we study computational problems related to rings and their automorphisms. We consider the problem GroupRA of computing the automorhism group of a given ring in terms of a set of generators of its automorphism group. We show that an efficient deterministic algorithm for GroupRA would imply the existence of efficient deterministic algorithms for a number of well-studied problems of intermediate complexity including polynomial factoring (over finite fields), Integer factoring and Graph Isomorphism. On the other hand, we upper bound the complexity of GroupRA by showing that GroupRA is in the complexity class fnAM and therefore is not NP-hard (NP$\not\subseteq$ P $^{\text{GroupRA}}$) unless the polynomial hierarchy collapses.

We then consider the problem of computing a nontrivial automorphism of a given ring R and show that it is random-polynomial-time equivalent to integer factoring. We then investigate the complexity of determining the existence of a nontrivial automorphism of a given ring. This problem is shown to admit an efficient deterministic algorithm.

3.1 Introduction

Given a problem, can we realistically solve it using computers? Computational Complexity Theory aims to answer this question by describing the amount of resources required to solve the problem as a function of the input size. Those problems whose solution can be computed in time bounded by a polynomial in the size of the input are said to belong to the class P . By consensus, this is the class of problems considered efficiently solvable. Another intensely investigated class is the class NP consisting of search problems - problems for which the correctness of a given solution can be verified efficiently. An early breakthrough in complexity theory was the discovery of the phenomenon of NP-completeness [Kar72] and its wide applicability to naturally occuring computational problems [Coo71]. It involves the existence of certain problems in NP whose efficient solution would imply efficient solutions for all NP-problems. Since then, most natural problems in NP have been classified as either being NP-complete or in P . Only a handful of *natural* problems in NP remain unclassified.

This chapter is motivated by a common theme that underlies many of these unclassified problems. Most such unclassified problems have a strong algebraic and/or number-theoretic flavour. These problems include quadratic residuosity, integer factoring, polynomial factoring over finite fields and graph isomorphism. Even within these problems there appears to be a wide variation in hardness: integer factoring is believed to be average-case hard and therefore suitable for cryptographic purposes, Graph Isomorphism appears to be easy on the average but may be hard in the worst-case, whereas polynomial factoring is believed to be efficiently solvable. Somewhat surprisingly then is our first result that a common theme underlies these diverse problems. We show that these unclassified problems are manifestations of the underlying problem of computing the automorphism group of a given ring. Indeed many algorithms for these problems either explicitly or implicitly make use of the automorphisms of the corresponding ring. Thus it is also natural to independently study the complexity of computing the automorphism group of a given ring.

We will restrict our attention to finite rings with unity. We assume that the rings are given in terms of the *basis* of their additive group and the multiplication table of basis elements (see section 2.1.6 for details).

In this chapter we consider the problem GroupRA of computing the automorphism group of a given ring. More precisely, GroupRA is the following computational problem - given an input ring $(R, +, \cdot)$, output a set of automorphisms $(\phi_1, \phi_2, \ldots, \phi_d)$ which generate the automorphism group, $Aut(R)$ of the ring R. We show that polynomial factoring (over finite fields), quadratic residuosity, integer factoring and graph isomorphism all reduce to GroupRA. [1]

We then consider a decision version of GroupRA. The *ring automorphism* problem is

[1] The reduction of Graph Isomorphism to GroupRA is based on a construction of Saxena [Sax06].

to test if a ring has a non-trivial automorphism. We prove that this problem is in P . This is in contrast to the corresponding problem for graphs whose status is still open. On the other hand we show that the problem of *finding a nontrivial automorphism* of a given ring is equivalent to integer factoring. This implies that the search version of the problem is likely to be strictly harder than the decision version. The results of this chapter mostly appear in [KS05].

Remark. Before proceeding, we make two important disclaimers.

(i) Not all known natural algebraic/number theoretic problems of 'intermediate complexity' reduce to the problem of computing the automorphism group of a ring. The discrete logarithm problem, a very important and well-studied number-theoretic problem in the class NP $\cap$ coNP does not fall under this framework.

(ii) Our algorithm for determining the existence of a nontrivial automorphism of a given ring unfortunately does not imply an efficient algorithm for the corresponding problem for graphs - determine if there exists a a nontrivial automorphism of a given graph. We will expand on this remark later.

3.2 The output of GroupRA.

An algorithm for GroupRA is expected to output a set of automorphisms generating the automorphism group of a ring R. It is therefore natural to ask whether there exists a small-sized generating set and whether we can efficiently verify if each of the output maps is an automorphism or not. In this section we first observe that there does indeed exist a generating set of size polynomial in $(\log|R|)$. We then show that we can verify if a map $\phi : (R, +) \mapsto (R, +)$ is an automorphism of R or not.

The following proposition argues that the automorphism group G of a ring R can have size at most exponential in $(\log|R|)$ and consequently there exists a generating set of G of size poly$((\log|R|))$.

Proposition 3.2.1. *The automorphism group G of any finite ring R has a generating set of size* $(\log|R|)^2$.

Proof. The size of a basis of $(R, +)$ is at most $n = \log|R|$. Now any homorphism $(R, +) \mapsto (R, +)$ is completely described by describing the images of all the basis elements and consequently there are at most $|R|^n$ such maps. Any automorphism is in particular a homomorphism from $(R, +)$ to $(R, +)$ and consequently the number of automorphisms, $\#G$ of R is also bounded by $|R|^n$.

Now from a sequence of automorphisms generating G one can eliminate redundancies - automorphisms which lie in the subgroup of G generated by the automorphisms occuring previously. In this way, we obtain a sequence of *irredundant* automorphisms so that the subgroup of G generated by the first i elements in the sequence is a proper subgroup of the group generated by the first $(i + 1)$ elements. By Lagrange's theorem for groups the size of the subgroup generated by the first $(i + 1)$ elements is at least twice as large the size of the subgroup of G generated by the first i elements. Consequently the length of an irredundant sequence of generators is at most $\log|G| \leq (\log|R|)^2$, as required.

□

Proposition 3.2.2. *Given a ring $(R, +, .)$ and a map $\phi : (R, +) \mapsto (R, +)$ we can verify in deterministic polynomial time if $\phi \in Aut(R)$.*

Proof. Let us assume that the additive group of the ring R is provided as:

$$(R, +) = (\mathbb{Z}/m_1\mathbb{Z})b_1 \oplus \ldots \oplus (\mathbb{Z}/m_n\mathbb{Z})b_n,$$

where $b_1, \ldots, b_n$ form a basis of the additive group $(R, +)$.

Moreover in the description of R we are given integers $((a_{i,j,k}))_{i,j,k\in[n]}$ such that

$$b_i \cdot b_j = \sum_{k=1}^{n} a_{i,j,k} b_k.$$

ϕ is an automorphism of R iff it satisfies the following conditions:

- *ϕ preserves addition*: check whether for all $1 \leq i \leq n,\ \ m_i \cdot \phi(b_i) = 0$.
- *ϕ preserves multiplication*: check whether for all $1 \leq i, j \leq n,\ \phi(b_i) \cdot \phi(b_j) = \sum_{k=1}^{n} a_{i,j,k}\phi(b_k)$, where $((a_{i,j,k}))_{i,j,k\in[n]}$ is the same matrix as given in the description of R.
- *ϕ is an invertible map from $(R, +)$ to $(R', +)$*: check whether $det(A) \in (\mathbb{Z}/(m_1m_2 \ldots m_n)\mathbb{Z})^*$, where A is the $n \times n$ integer matrix describing the map $\phi : R \to R'$.

The first two conditions above imply that ϕ is a homomorphism between the two rings. The third condition ensures that ϕ is bijective. All these three conditions can be checked in polynomial time. □

3.3 Lower Bounds for GroupRA

In this section we show that a number of well-studied computational problems of *intermediate complexity* are deterministically reducible to the problem of computing the automorphism group of a given ring. In these reductions we will be using GroupRA as an oracle and then using this oracle to solve various problems in deterministic polynomial time. It is therefore natural to first investigate the elementary computations that one can do using the automorphism group of a ring.

3.3.1 Elementary Operations

Let $(R, +, \cdot)$ be a finite commutative ring with identity and $Aut(R)$ be its automorphism group. Suppose further that the automorphism $\phi_1, \ldots, \phi_d \in Aut(R)$ generate $Aut(R)$. For an element $r \in R$, define the orbit of r denoted Orbit_r to be the set $\{\phi(r) | \phi \in Aut(R)\}$. We first observe that if we are given the generators for $Aut(R)$ then we can efficiently compute the orbit of every element of R.

Now suppose that H is a subgroup of $Aut(R)$. Then associated with H is a subring R^H of R consisting of elements of r which are fixed by every automorphism in H. That is,

$$R^H \stackrel{\text{def}}{=} \{r \in R \mid \forall \psi \in H : \psi(r) = r\}$$

R^H is then called the subring of R fixed by H. We will also see that given a set of generators for the subgroup H of $Aut(R)$, the computation of R^H boils down to solving a set of linear equations and hence can be done efficiently.

▮ *Computing the orbit of an element.*

The next proposition shows that given the generators of the automorphism group we can efficiently compute any required number m of elements in the orbit of an element of R.

Proposition 3.3.1. *Suppose we are given an input ring R and its automorphism group in terms of a set of generators $\{\phi_1, \ldots, \phi_d\}$, then for every $r \in R$, we can compute m elements in its orbit, if as many exist, in deterministic time poly$(m \cdot \log |R|)$.*

Proof: Fix the $r \in R$ whose orbit we desire to compute. Define the sets

$$A_0 \subseteq A_1 \subseteq \ldots \subseteq \text{Orbit}_r$$

inductively as follows:

- $A_0 \overset{\text{def}}{=} \{r\}$.
- $A_{i+1} \overset{\text{def}}{=} A_i \cup \bigcup_{j=1}^{j=d} \{\phi_j(x) | x \in A_i\}$

It is easy to verify that for all $i \geq 0$,

$$A_i \subseteq A_{i+1} \subset \text{Orbit}_r$$

and that the elements of A_{i+1} can be computed efficiently given the elements of A_i. Moreover, if at some stage t we have $A_{t+1} = A_t$ then $\text{Orbit}_r = A_t = A_{t+1}$.

Now given an integer $m \geq 1$, we succesively compute $A_0, A_1, A_2, \ldots$ until we reach a set A_t such that either $|A_t| \geq m$ or $A_t = \text{Orbit}_r$. If $A_t = Orbit_r$ then we have computed the entire orbit of r and we can output accordingly. Else we have computed at least m elements in the orbit and we output the first m elements of A_t.

▮

▮ *Computing the subring fixed by a subgroup.*

We next show that given a sequence of automorphisms $\psi_1, \ldots, \psi_d \in Aut(R)$ generating a subgroup H of $Aut(R)$ we can efficiently compute the subring R^H of R which is fixed by H. This happens because the computation of R^H boils down to simply solving a set of homogeneous linear equations over $\mathbb{Z}/m\mathbb{Z}$, where m is the characteristic of R.

Proposition 3.3.2. *Given a ring R and a set of automorphisms $\psi_1, \ldots, \psi_d \in Aut(R)$ generating a subgroup H of $Aut(R)$, we can efficiently compute the subring R^H of R which is pointwise fixed by each of these automorphisms.*

Proof. Let us assume that the additive group of the ring R is provided as:

$$(R, +) = (\mathbb{Z}/m_1\mathbb{Z})b_1 \oplus \ldots \oplus (\mathbb{Z}/m_n\mathbb{Z})b_n.$$

Then $m = \text{Lcm}(m_1, \ldots, m_n)$ is the characteristic of R.

Now observe that an element

$$\mathbf{x} \in (R, +),\ \mathbf{x} = x_1 b_1 + x_2 b_2 + \ldots + x_n b_n$$

is in R^H if and only if $\psi(\mathbf{x}) = \mathbf{x}$ for all $\psi \in \{\psi_1, \ldots, \psi_d\}$. This amounts to a set of $(d \cdot n)$ homogeneous linear equations in the unknowns x_i's over the ring $\mathbb{Z}/m\mathbb{Z}$. Solving this set of linear equations we obtain a basis of the additive group of R^H.

Finally, expressing the product of each pair of basis elments of $(R^H, +)$ as a linear combination of the basis elements we obtain the complete description of R^H.

□

3.3.2 Decomposing a ring using GroupRA.

We show here that GroupRA can be used to factor integers and polynomials over finite fields. Both these factoring problems are special cases of the problem of decomposing a ring into its component local rings. We show that GroupRA can in fact be used to completely decompose a ring. The key idea in this decomposition is the notion of an *idempotent* element which is an element $e \in R$ satisfying $e^2 = e$. We will see that a nontrivial idempotent $(e \neq 0, 1)$ $e \in R$ leads to the following decomposition of R:

$$R = R \cdot e \otimes R \cdot (1 - e)$$

Definition 3.3.3. Let R be a ring. An element $e \in R$ is said to be an *idempotent* element if and only if e satisfies $e^2 = e$. An idempotent e is said to be *nontrivial* if and only if $e \neq 0, 1$.

Lemma 3.3.4. *[McD74] A commutative ring R is decomposable if and only if it contains a nontrivial idempotent element. Moreover if $e \in R$ is a nontrivial idempotent then*

$$R = R \cdot e \otimes R \cdot (1 - e)$$

is a decomposition of R.

Proof. **($\Longrightarrow$).** Suppose that

$$R = R_1 \otimes R_2$$

Then the elements $(1, 0)$ and $(0, 1)$ are nontrivial idempotents of R.

($\Longleftarrow$). Let e be a nontrivial idempotent of R. Then observe that $(1 - e) \neq e$ is also a nontrivial idempotent of R. Also observe that the idempotence of e implies that the set

$$R \cdot e \overset{\text{def}}{=} \{r \cdot e,\ r \in R\}$$

is the same as the set $\{r \in R \mid r \cdot e = r\}$ and is a subring of the ring R with the element e as its multiplicative identity. That is

$$R \cdot e \overset{\text{def}}{=} \{r \cdot e,\ r \in R\} = \{r \in R \mid r \cdot e = r\} \tag{3.1}$$

Similarly, the set

$$R \cdot (1-e) \overset{\text{def}}{=} \{r \cdot (1-e),\ r \in R\} = \{r \in R \mid r \cdot (1-e) = r\} \tag{3.2}$$

is a subring of R containing $(1-e)$ as its multiplicative identity. We claim that the required decomposition of R is then given by

$$R = R \cdot e \otimes R \cdot (1-e).$$

To prove this we need to show three things

(i). $R \cdot e$ and $R \cdot (1-e)$ are both ideals of R.

Proof. To see that $R \cdot e$ is an ideal of R observe that for any $a \in R$, $a \cdot (r \cdot e) = (a \cdot r) \cdot e \in R \cdot e$ and hence $a \cdot (R \cdot e) \subseteq R \cdot e$ for all $a \in R$. A similar argument shows that $R \cdot (1-e)$ is an ideal of R. □

(ii). $R \cdot e \cap R \cdot (1-e) = \{0\}$.

Proof. Suppose that $r \in R \cdot e \cap R \cdot (1-e)$. Then by equations 3.1 and 3.2 we have $r \cdot e = r$ and $r \cdot (1-e) = r$. Together they imply that $r = 0$. □

(iii). Every element $r \in R$ can be uniquely written as $r = r_1 + r_2$ where $r_1 \in R \cdot e$ and $r_2 \in R \cdot (1-e)$.

Proof. For any element $r \in R$ we have $r = (r \cdot e) + (r \cdot (1-e))$ and therefore r can certainly be expressed as the sum of an element in $R \cdot e$ and an element in $R \cdot (1-e)$. Further suppose that there exist two different representations of this form of an element $r \in R$. If

$$r = r_1 + r_2 = r_1' + r_2',$$

then we have

$$r_1 - r_1' = r_2 - r_2'.$$

But $(r_1 - r_1') \in R \cdot e$ and $(r_2 - r_2') \in R \cdot (1-e)$ so that by (ii) above we have $r_1 - r_1' = r_2 - r_2' = 0$. Thus such a representation of r is unique. □

This completes the proof of the lemma.

□

Remark. Oracles for integer factoring and polynomial factoring over finite fields can be used to find nontrivial idempotents in a ring R and thereby compute its complete decomposition. To see this, suppose that for two coprime integers d_1 and d_2, R contains $(\mathbb{Z}/d_1d_2\mathbb{Z})$ as a subring, then the element $e \overset{\text{def}}{=} (d_1^{-1} \pmod{d_2})d_1$ is a nontrivial idempotent in $(\mathbb{Z}/d_1d_2\mathbb{Z})$ and hence also a nontrivial idempotent in R. Similarly for two coprime polynomials $f(x), g(x) \in (\mathbb{Z}/p^\alpha\mathbb{Z})[x]$, if the ring $(\mathbb{Z}/p^\alpha\mathbb{Z})[x]/\langle f(x) \cdot g(x)\rangle$ is a subring of R then the element $e \overset{\text{def}}{=} (f(x)^{-1} \pmod{g(x)}) \cdot f(x)$ is a nontrivial idempotent in $(\mathbb{Z}/p^\alpha\mathbb{Z}[x])/\langle f(x) \cdot g(x)\rangle$ and hence in R.

We now show that a GroupRA oracle can be used to deterministically compute a nontrivial idempotent of a given ring R and thereby decompose a given ring R into its indecomposable or local subrings.

Proposition 3.3.5. *Using an oracle for GroupRA we can compute the decomposition of a given ring R in deterministic polynomial time.*

Proof. Let S be the ring

$$S \overset{\text{def}}{=} R[z]/\langle z^2 - z\rangle.$$

The next claim shows that the elements in the orbit of z under $Aut(s)$ correspond correspond to idempotents of R. It is easy to verify the following claim.

Claim 3.3.5.1. *If $e \in R$ is an idempotent of R then the map $\phi : S \mapsto S$, $z \mapsto e \cdot (1-z) + (1-e) \cdot z$ is an automorphism of S. In the converse direction, if $\phi \in Aut(S)$ is an automorphism of S, $\phi : z \mapsto (a + bz)$ the a is an idempotent of R. Further if $(a + bz) \notin \{z, 1-z\}$ then a is a nontrivial idempotent of R.*

This means that R is decomposable if and only if S contains an automorphism $\phi : S \mapsto S$ such that $\phi(z)$ is different from z and $(1-z)$. Thus by proposition 3.3.1, using an oracle for GroupRA, we can efficiently compute 3 distinct elements in the orbit of the element $z \in S$. At least one of them gives us a non-trivial idempotent e of R. We then use lemma 3.3.4 to decompose R into two component subrings $R = R \cdot e \otimes R \cdot (1-e)$. Indeed if $b_1, \ldots, b_n$ form a basis of the additive group $(R, +)$ of R, then every element in $R \cdot e$ can be expressed as an integer linear combination of the elements $b_1 \cdot e, \ldots, b_n \cdot e \in R \cdot e$. Eliminating redundancies gives us a basis of the component subring $R \cdot e$. Expressing the product of any two basis elements as an integer linear combination of these basis elements gives us a complete description of the ring $R \cdot e$. Similarly we compute a complete description of the subring $R \cdot (1 - e)$.

Having decomposed R into two subrings we recursively decompose these two subrings to obtain a complete decomposition of R in deterministic polynomial time.

□

As a corollary of this we get that both integer factoring and polynomial factoring over finite fields reduce to GroupRA.

Corollary 3.3.6. *Polynomial Factoring $\leq_T^P$ GroupRA.*

Proof. It is well-known (cf. Berlekamp [Ber70]) that over any field, general polynomial factorization problem reduces to factoring square-free polynomials. Now given a finite field $\mathbb{F}_q$ and a square-free polynomial $f(x) \in \mathbb{F}_q[x]$ having factorization

$$f(x) = \prod_{i=1}^{m} f_i(x),$$

the ring $R \overset{\text{def}}{=} \mathbb{F}_q[x]/\langle f(x)\rangle$ has the unique decomposition

$$R \cong \bigotimes_{i=1}^{m} \mathbb{F}_q[x]/\langle f_i(x)\rangle.$$

By Proposition 3.3.5 we can compute this decomposition of R using an oracle for GroupRA together with projection maps $\pi_i : R \mapsto \mathbb{F}_q[x]/\langle f_i(x)\rangle$. Computing the minimum polynomial of $\pi_i(x)$ in each of the component subrings gives us the desired factors $f_i(x)$ of $f(x)$.

□

In a similar vein, we have

Corollary 3.3.7. *Integer Factoring $\leq_T^P$ GroupRA.*

Proof. Given an integer n having prime factorization

$$n = \prod_{i=1}^{m} p_i^{e_i},$$

the ring $\mathbb{Z}/n\mathbb{Z}$ has the unique decomposition

$$\mathbb{Z}/n\mathbb{Z} \cong \bigotimes_{i=1}^{m} \mathbb{Z}/p_i^{e_i}\mathbb{Z}.$$

By Proposition 3.3.5 we can compute this decomposition of $\mathbb{Z}/n\mathbb{Z}$ and thereby obtain the prime factorization of n. □

It is well-known that Quadratic Residuosity $\leq_T^P$ Integer Factoring (cf. [BGry]) and thus we get the following corollary.

Corollary 3.3.8. *Quadratic Residuosity (modulo composites) $\leq_T^P$ GroupRA.*

3.3.3 Ring Isomorphism testing reduces to GroupRA.

In this section we show that an oracle for GroupRA can be used to efficiently decide if two given finite commutative rings are isomorphic or not. We use this result, together with a construction of Saxena [Sax06] in the next subsection to show that the Graph Isomorphism problem also efficiently reduces to GroupRA.

First we show how to use the GroupRA oracle to determine if two given local rings are isomorphic.

Lemma 3.3.9. *Let R_1 and R_2 be two commutative local rings. Let $R \overset{\text{def}}{=} R_1 \otimes R_2$. Then the orbit of the element $(1,0) \in R$ under $Aut(R)$ contains two distinct elements if and only if R_1 and R_2 are isomorphic.*

Proof. Observe that an element $(r_1, r_2) \in R$ is an idempotent of R if and only if $r_1 \in R_1$ and $r_2 \in R_2$ are idempotents in their respective rings. By assumption R_1 and R_2 are commutative local rings and therefore contain no nontrivial idempotents. Thus the only nontrivial idempotents in R are $(1,0)$ and $(0,1)$. Since nontrivial idempotents may contain only nontrivial idempotents in their orbit therefore the orbit of the element $(1,0) \in R$ is either $\{(1,0),(0,1)\}$ or just $\{(1,0)\}$.

($\Longrightarrow$.) If $(0,1) \in \text{Orbit}_{(1,0)}$ then there is an automorphism $\phi \in Aut(R)$ such that $\phi((1,0)) = (0,1)$. Now we have

$$\begin{aligned}\phi((r_1,0)) &= \phi((r_1,0)\cdot(1,0))\\ &= \phi((r_1,0))\cdot\phi((1,0))\\ &= \phi((r_1,0))\cdot(0,1)\\ &= (0,r_2) \text{ for some } r_2 \in R_2.\end{aligned}$$

Thus ϕ induces an isomorphism $\phi' : R_1 \mapsto R_2$, $\phi' : r_1 \mapsto r_2$ where r_2 is the unique element of R_2 such that $\phi(r_1, 0) = (0, r_2)$. It is easy to verify that ϕ' is indeed an isomorphism from R_1 to R_2.

($\Longleftarrow$.) Suppose $\phi : R_1 \mapsto R_2$ is a ring isomorphism. Then ϕ induces an automorphism

$$\phi' : R \mapsto R, \;\; \phi' : (r_1, r_2) \mapsto (\phi^{-1}(r_2), \phi(r_1)).$$

So we have

$$\begin{aligned}\phi'(1,0) &= (\phi^{-1}(0), \phi(1)) \\ &= (0,1) \text{ [since } \phi(0)=0 \text{ and } \phi(1)=1]\end{aligned}$$

and thus $(0,1) \in \mathrm{Orbit}_{(1,0)}$ as required. □

Thus given two local rings R_1 and R_2 whose isomorphism we want to test, we form the ring $R \overset{\text{def}}{=} R_1 \otimes R_2$ and by Proposition 3.3.1, we can efficiently compute the orbit of the element $(1,0) \in R$ using an oracle for GroupRA and thereby determine if R_1 is isomorphic to R_2 or not using the lemma 3.3.9 above. This gives us the following lemma.

Lemma 3.3.10. *Given two finite commutative local rings R_1 and R_2 as input, we can efficiently determine if R_1 is isomorphic to R_2 using an oracle for GroupRA. That is, Local Ring Isomoprhism $\leq_T^P$ GroupRA.*

This lemma can be generalized to show that indeed the ring isomorphism problem, of testing whether two rings are isomorphic or not efficiently reduces to GroupRA.

Theorem 3.3.11. *Given two finite commutative rings R_1 and R_2 as input, we can efficiently determine if R_1 is isomorphic to R_2 using an oracle for GroupRA. That is, Ring Isomoprhism $\leq_T^P$ GroupRA.*

Proof. Let the decomposition of R_1 and R_2 into their component local rings be

$$R_1 = \bigotimes_{i=1}^{m_1} R_{1i}, \quad R_2 = \bigotimes_{i=1}^{m_2} R_{2i}. \tag{3.3}$$

By Proposition 3.3.5, an oracle for GroupRA can be used to compute the decomposition 3.3 of R_1 and R_2. Then R_1 is isomorphic to R_2 if and only if the number of component local rings are the same ($m_1 = m_2$) and every component local ring R_{1i} of R_1 is isomorphic to some corresponding local ring R_{2j} of R_2. Thus isomorphism testing of R_1 and R_2 now boils down to isomorphism testing of the component local rings R_{1i}'s and R_{2j}'s, which can be done efficiently by lemma 3.3.10.

□

Remark: The structure theorem for finite abelian groups can be used to check in polynomial time whether for two rings, given in basis form, the additive groups are isomorphic or not. Suppose the two additive groups are $G := (\mathbb{Z}/d_1\mathbb{Z}) \oplus \cdots \oplus (\mathbb{Z}/d_n\mathbb{Z})$ and

$G' := (\mathbb{Z}/d'_1\mathbb{Z}) \oplus \cdots \oplus (\mathbb{Z}/d'_n\mathbb{Z})$. Consider the set $D = \{d_i \mid \ i \in [n]\} \cup \{d'_i \mid \ i \in [n]\}$. We take *gcd*s of all pairs of integers from the set D and expand D in each such *gcd*-operation as: if $\alpha,\ \beta \in D$ have a nontrivial *gcd* then replace them by $\frac{\alpha}{gcd(\alpha,\beta)}$, $\frac{\beta}{gcd(\alpha,\beta)}$ and $gcd(\alpha,\beta)$. We can keep repeating this process on the new expanded D till all the elements of D become mutually coprime. It is guaranteed to stop in polynomial time, for D can expand to a maximum size of $\log(\#G \cdot \#G')$ as the number of prime factors of a number N are less than $\log N$. Now factor d_i 's and d'_j 's as much as possible using the numbers from D. Say, $d_i = d_{i,1}^{e_1} \cdots d_{i,k}^{e_k}$ where $d_{i,1}, \ldots, d_{i,k} \in D$ are mutually coprime. We can refine the decomposition of G by breaking $(\mathbb{Z}_{d_i}, +)$ as:

$$(\mathbb{Z}/d_{i,1}^{e_1}\mathbb{Z}) \oplus \cdots \oplus (\mathbb{Z}/d_{i,k}^{e_k}\mathbb{Z}).$$

At the end of all this refining of d_i's and d'_j's using D, let the *finer* structural decompositions be: $G \cong (\mathbb{Z}/m_1\mathbb{Z}) \oplus \cdots \oplus (\mathbb{Z}/m_{n'}\mathbb{Z})$ and $G' \cong (\mathbb{Z}/m'_1\mathbb{Z}) \oplus \cdots \oplus (\mathbb{Z}/m'_{n'}\mathbb{Z})$. Now by invoking the structure theorem: G will be isomorphic to G' if and only if the *multi-sets* (i.e. elements with repetition) $\{m_i\}_{i\in[n']}$ and $\{m'_i\}_{i\in[n']}$ are equal. ∎

3.3.4 Graph Isomorphism reduces to GroupRA.

In this subsection our aim is to show that the Graph Isomorphism problem deterministically reduces to GroupRA. It is based on a construction of Saxena [Sax06]. Corresponding to a graph G, the construction of Saxena [Sax06] gives a local $\mathbb{F}_5$-algebra $R(G)$ such that two such local algebras $R(G_1)$ and $R(G_2)$ are isomorphic if and only if their source graphs G_1 and G_2 are isomorphic. We reproduce the construction and for the sake of completeness, give the proof in the appendix.

Let G be an undirected graph with n vertices and no self loops. The construction involves a local commutative $\mathbb{F}_5$-algebra. Associate variables to each vertex (x-variable) and capture the "connectivity" of the graph by defining the edges-polynomial – $\sum_{(u,v)\text{ is an edge}} x_u x_v$ – as zero in the ring.

Define the following commutative $\mathbb{F}_5$-algebra:

$$R(G) := \mathbb{F}_5[x_1, \ldots, x_n]/\mathcal{I}$$

where, ideal $\mathcal{I}$ has the following relations:

1. x's are nilpotents of degree 2, i.e., for all $i \in [n]$: $x_i^2 = 0$.

2. the edges-polynomial is zero, i.e., $\sum_{\substack{1\leq i<j\leq n \\ (i,j)\in E(G)}} x_i x_j = 0$.

3. all cubic terms are zero, i.e., for all $i, j, k \in [n] : \ x_i x_j x_k = 0$.

Suppose (i_0, j_0) is an edge in G such that $1 \leq i_0 < j_0 \leq n$. Then the additive structure of the ring is:

$$(R(G), +) = \mathbb{F}_5 \cdot 1 \oplus \bigoplus_{i\in[n]} \mathbb{F}_5 \cdot x_i \oplus \bigoplus_{\substack{i<j\in[n] \\ (i,j)\neq(i_0,j_0)}} \mathbb{F}_5 \cdot (x_i x_j)$$

Thus, the dimension of the ring over $\mathbb{F}_5$ is $\binom{n+1}{2}$. Multiplication satisfies the associative law simply because the product of any three *variables* (in any order) is zero. Also, $R(G)$ is a local commutative $\mathbb{F}_5$-algebra.

Observe that if $G_1 \cong G_2$ then any graph isomorphism ϕ induces a natural isomorphism between rings $R(G_1)$ and $R(G_2)$. So we only have to prove the converse:

Lemma 3.3.12. *[Sax06]. Let G_1 and G_2 be two undirected graphs having no self-loops. Further, assume that graphs G_1 and G_2 are not a disjoint union of a clique and a set of isolated vertices. Then, $R(G_1) \cong R(G_2)$ implies $G_1 \cong G_2$.*

Proof. See the appendix. □

It is easy to see that testing isomorphism of arbitary graphs reduces to testing isomorphism of connected graphs. Consequently in what follows we will work with simple undirected connected graphs. The lemma above essentially implies that Graph Isomorphism reduces to the problem of testing the isomorphism of two local rings. By Theorem 3.3.11, this can be decided efficiently using an oracle for GroupRA and thus we have:

Theorem 3.3.13. *Graph Isomoprhism $\leq_T^P$ GroupRA.*

Since Graph Automorphism is turing-reducible to Graph Isomorphism (cf. [KST93]), we also get

Corollary 3.3.14. *Graph Automorphism $\leq_T^P$ GroupRA.*

3.4 Upper bounds for GroupRA.

In this section we give upper bounds for GroupRA and show that GroupRA is in the complexity class fnAM. By Proposition 2.2.3, this implies that if GroupRA is NP-hard (i.e.

NP $\subseteq$ P $^{\text{GroupRA}}$) then PH collapses to Σ_2, an ‘unlikely’ event. Let us first upper bound the complexity of the closely related problem of counting the number of automorphisms of a given ring. We will subsequently use this to derive our upper bound for GroupRA.

3.4.1 The Complexity of Counting Ring Automorphisms.

In this subsection we consider the counting version of the ring automorphism problem.

Definition 3.4.1. The computational problem #RA is defined as the functional problem of *computing the number of automorphisms* of a given ring. Its decision version is the language:

$$\text{cRA} := \{(R, k) \mid R \text{ is a ring in basis form s.t. } \#Aut(R) \geq k\} \tag{3.4}$$

We begin the investigation of the complexity of #RA by investigating the complexity of counting the number of automorphisms of the additive group $(R, +)$ of the given ring R.

▮ *Counting the number of automorphisms of* $(R, +)$.

Using the structure theorem of abelian groups, we can compute $\#Aut(R, +)$ of a ring R presented in terms of additive generators having prime-power additive orders.

Proposition 3.4.2. *Given a ring R in terms of additive generators, all having prime-power additive orders, we can compute the number of automorphisms of the additive group of R, $\#Aut(R, +)$, in polynomial time.*

Proof. Automorphisms of the additive group $(R, +)$ are nothing but the invertible linear maps on the additive generators of R. Thus, to compute $\#Aut(R, +)$ we compute the number of invertible linear maps or the number of invertible matrices.

Let $(R, +)$ be given as $\cong \bigoplus_{i=1}^{l} \bigoplus_{j} (\mathbb{Z}/{p_i}^{\alpha_{i,j}}\mathbb{Z})$, where p_i's are distinct primes and $\alpha_{i,j} \in \mathbb{Z}^{\geq 1}$. For $1 \leq i \leq l$ define subrings R_i of R as:

$$R_i := \{r \in R \mid r \text{ has power-of-}p_i \text{ additive order}\}$$

Observe that

$$R \cong R_1 \times \cdots \times R_l$$

this is because if $r_i \in R_i$ and $r_j \in R_j$ $(i \neq j)$ then for some $c_i, c_j \in \mathbb{Z}^{\geq 0}$, $p_i^{c_i} r_i r_j = p_j^{c_j} r_i r_j = 0$ which implies that $r_i r_j = 0$ (since $\exists a, b \in \mathbb{Z}$ such that $a p_i^{c_i} + b p_j^{c_j} = 1$) and by a similar argument $r_1 \in R_1, \ldots, r_l \in R_l$ are *linearly independent.*

This decomposition of R gives us:

$$\#Aut(R,+) = \prod_{i=1}^{l} \#Aut(R_i,+)$$

Thus, it suffices to show how to compute $\#Aut(R,+)$ when $(R,+)$ is given as $\cong \bigoplus_{i=1}^{n}(\mathbb{Z}/p^{\alpha_i}\mathbb{Z})$ where p is a prime and $\alpha_i \in \mathbb{Z}^{\geq 1}$.

Suppose we are given R in terms of the following additive basis:

$$\begin{aligned}(R,+) \quad &= \quad (\mathbb{Z}/p^{\beta_1}\mathbb{Z})e_{1,1} \oplus \ldots \oplus (\mathbb{Z}/p^{\beta_1}\mathbb{Z})e_{1,n_1} \oplus \ldots \\ &\quad\quad \ldots \oplus (\mathbb{Z}/p^{\beta_m}\mathbb{Z})e_{m,1} \oplus \ldots \oplus (\mathbb{Z}/p^{\beta_m}\mathbb{Z})e_{m,n_m}\end{aligned}$$

where, $n_1 + \ldots + n_m = n$ and $1 \leq \beta_1 < \ldots < \beta_m$.

Observe that $\phi \in Aut(R,+)$ iff the matrix A describing the map ϕ is invertible (mod p) and preserves the additive orders of $e_{i,j}$'s. Our intention is to count the number of all such matrices A. To do that let us see how A looks:

$$A = \begin{pmatrix} B_{1,1} & B_{1,2} & \ldots & B_{1,m} \\ B_{2,1} & B_{2,2} & \ldots & B_{2,m} \\ \vdots & \ldots & \ddots & \vdots \\ B_{m,1} & B_{m,2} & \ldots & B_{m,m} \end{pmatrix}_{n\times n}$$

where the block matrices $B_{i,j}$'s are integer matrices of size $n_i \times n_j$. The properties of these block matrices which make A describe an automorphism of $(R,+)$ are:

- for $1 \leq j < i \leq m$: entries in $B_{i,j}$ are from $\{0,1,\ldots,p^{\beta_j}-1\}$.
- for $1 \leq i \leq m$: entries in $B_{i,i}$ are from $\{0,1,\ldots,p^{\beta_i}-1\}$ and $B_{i,i}$ is invertible (mod p).
- for $1 \leq i < j \leq m$: entries in $B_{i,j}$ are from $\{0,1,\ldots,p^{\beta_j}-1\}$ and $B_{i,j} \equiv 0 \pmod{p^{\beta_j-\beta_i}}$.

It is not difficult to see that the number of matrices satisfying these conditions can be found in time polynomial in $(n_1\beta_1 + \ldots + n_m\beta_m)(\log p)$, and hence the number of A's which describe an automorphism of $(R,+)$.

□

Remark: When a ring R is given in terms of generators having composite additive orders then computing $\#Aut(R,+)$ entails factoring integers. For example, suppose $n = pq$ where $p \neq q$ are primes and ring R is given as $(\mathbb{Z}/n\mathbb{Z}, +, \cdot)$. Then $\#Aut(R,+) = (p-1)(q-1) = \phi(n)$ and if we compute $\phi(n)$ then we can factorize n in randomized polynomial time (see [Mil76]). ▮

This section will explore the complexity of the problem of counting ring automorphisms. We will show that the natural decision version of the counting problem, cRA is in the complexity class AM $\cap$ coAM and therefore is unlikely to be NP-hard.

▮ *cRA* $\in$ *AM* $\cap$ *coAM* .

We will now show that given a finite ring R there is an AM protocol in which Merlin sends a number ℓ and convinces Arthur that $\#Aut(R) = \ell$. The ideas in the proof are basically from Babai and Szemeredi [BS84].

Theorem 3.4.3. $\#RA \in FP^{\,AM \cap coAM}$.

Proof. Let R be a finite ring given in its basis form. We will first show how Merlin can convince Arthur that $\#Aut(R) \geq k$. Recall that in Equation (3.4) we defined this problem as cRA.

Claim 3.4.3.1. *cRA* $\in$ *AM* .

Proof of Claim 3.4.3.1. Merlin can give Sylow subgroups $S_{p_1}, \ldots, S_{p_m}$ of $Aut(R)$, in terms of generators, to Arthur such that $p_1, \ldots, p_m$ are distinct primes and the product $|S_{p_1}|. \cdots .|S_{p_m}| \geq k$. Arthur now has to verify whether for a given Sylow subgroup S_p, $|S_p| = p^t$ or not. So Merlin can further provide the composition series of S_p:

$$S_p = G_t > G_{t-1} > \ldots > G_1 > G_0 = \{1\}.$$

Suppose, by induction, that Arthur is convinced about $|G_i| = p^i$. Then to prove $|G_{i+1}| = p^{i+1}$, Merlin will provide $x_{i+1} \in G_{i+1}$ to Arthur with the claim that $x_{i+1} \notin G_i$ but $x_{i+1}^p \in G_i$. Latter can be verified easily by Arthur as Merlin can give the way to produce x_{i+1}^p from the generators of G_i. Finally, the only nontrivial thing left for Arthur to verify is whether $x_{i+1} \notin G_i$, which can be verified by a standard AM protocol (Proposition 2.2.1) as there is a gap in the size of the set $X :=$ (group generated by x_{i+1} and G_i):

$$x_{i+1} \notin G_i \quad \Rightarrow \quad \#X = p^{i+1}$$
$$x_{i+1} \in G_i \quad \Rightarrow \quad \#X = p^{i}$$

To avoid too many rounds, Merlin first provides $x_0 = 1, x_1, \ldots, x_t \in Aut(R)$ with the proof of: for all $1 \leq i \leq t,\ x_i^p \in G_{i-1} :=$ (group generated by $x_0, \ldots, x_{i-1}$) to Arthur and then provides the proof of: for all $1 \leq i \leq t,\ x_i \notin G_{i-1}$ in the second round for Arthur to verify. □

Now we give the AM protocol that convinces Arthur of $\#Aut(R) \leq k$.

Claim 3.4.3.2. *$cRA \in coAM$.*

Proof of Claim 3.4.3.2. Arthur has a finite ring R and he wants a proof of $\#Aut(R) \leq k$. Firstly, Merlin provides the prime factorization of the characteristic of R which Arthur uses to decompose the given ring into component rings of prime-power order. The automorphism group of the ring R is then simply a direct product of the automorphism groups of the prime-power order subrings of R. Then the size of the automorphism group of R is the product of the sizes of the automorphism groups of the prime-power order subrings of R. Merlin now needs to prove an appropriate lower bound on the sizes of each of these prime-power subrings of R.

So now we can assume that R is given in terms of generators having prime-power additive orders. For concreteness let us assume:

$$(R, +) = \bigoplus_{i=1}^{n} (\mathbb{Z}/p_i^{\alpha_i}\mathbb{Z}) b_i$$

Merlin sends Arthur a number $\ell \leq k$ as a candidate value for $\#Aut(R)$ and also provides some Sylow subgroups, the product of their sizes being equal to ℓ, with the AM-proofs for their sizes (as used in Claim 3.4.3.1). Let

$$\begin{aligned} X \quad := \quad & \{ \left\langle ((a_{i,j,k}))_{i,j,k\in[n]} \right\rangle \mid \exists \pi \in Aut(R,+) \text{ s.t. } \pi(b_i) \cdot \pi(b_j) = \sum_{k=1}^{n} a_{i,j,k} \pi(b_k); \\ & \text{for all } 1 \leq i,j,k \leq n,\ 0 \leq a_{i,j,k} < p_k^{\alpha_k} \ \} \,. \end{aligned}$$

Observe that $\#X = \frac{\#Aut(R,+)}{\#Aut(R)}$ and $\#Aut(R,+)$ can be computed in polynomial time when $(R,+)$ is given in terms of generators having prime-power additive orders (see Proposition 3.4.2). Thus, Arthur computes $s := \#Aut(R,+)$. Arthur is already convinced that $\ell | \#Aut(R)$ and he now wants to verify $\#Aut(R) \leq \ell$. A standard AM protocol (see

Proposition 2.2.1) now follows by utilizing the gap in the size of X in the two cases:

$$\begin{aligned} \#Aut(R) \leq \ell \quad &\Rightarrow \quad \#X \geq \frac{s}{\ell}\ . \\ \#Aut(R) > \ell \quad &\Rightarrow \quad \#Aut(R) \geq 2\ell \quad [\because \#Aut(R) \text{ has a subgroup of size } \ell] \\ &\Rightarrow \#X \leq \frac{s}{2\ell}\ . \end{aligned}$$

□

The claims above show that #RA $\in$ FP $^{\text{cRA}} \subseteq$ FP $^{\text{AM} \cap \text{coAM}}$. □

3.4.2 GroupRA is in fnAM.

Note that the AM protocols that we give for #RA not only count the number of automorphisms but give a lot more information about the automorphism group. In fact, these AM protocols compute the full automorphism group of a ring R in terms of the generators of the Sylow subgroups of $Aut(R)$.

Corollary 3.4.4. *Function GroupRA $\in$ fnAM and hence is low for Σ_2.*

Proof. Let f be the function, corresponding to GroupRA, that maps a ring R (given in basis form) to the tuple $(\#Aut(R), Aut(R))$. Since cRA is in both AM and coAM there are deterministic polynomial time Turing Machines A and B, and positive constants c, d such that:

$$\begin{aligned} \#Aut(R) \leq k \quad \text{iff} \quad & \text{Prob}_{y \in \{0,1\}^{\log^c \#R}}[(\exists z \in \{0,1\}^{\log^c \#R})\ A(R,k,y,z) \text{ accepts}] \\ & \geq \left(1 - \frac{1}{2^{\log^d \#R}}\right) \\ \#Aut(R) \geq k \quad \text{iff} \quad & \text{Prob}_{y \in \{0,1\}^{\log^c \#R}}[(\exists z \in \{0,1\}^{\log^c \#R})\ B(R,k,y,z) \text{ accepts}] \\ & \geq \left(1 - \frac{1}{2^{\log^d \#R}}\right) \qquad (3.5) \end{aligned}$$

The parameter d above will be chosen large enough so that all the subsequent arguments go through. To show that $f \in$ fnAM we plan to run A and B in parallel. We can modify A slightly to A' by requiring that $A(R,k,y,z)$ outputs (ℓ, G) where, ℓ is the number and G is the group, given by the generators of the (intended) Sylow subgroups, as occurred in the proof of the Claim 3.4.3.2. It is easy to see that:

$$\begin{aligned} & f(R) = (m, H) \\ \Rightarrow \quad & \mathrm{Prob}_{y \in \{0,1\}^{2 \log^c \#R}}[(\exists \ell' z z' \in \{0,1\}^{3 \log^c \#R}), \text{ both } A'(R, \ell', y, z) \\ & \quad \text{and } B(R, \ell', y, z') \text{ accept and } A'(R, \ell', y, z) = (m, H)] \geq \frac{3}{4} \end{aligned} \tag{3.6}$$

The above holds because Merlin can simply send ℓ' as equal to $\#G$ and a part of the string z and z' having the group $Aut(R)$ in terms of the generators of Sylow subgroups (see the proof of Claim 3.4.3.2). Then Equations (3.5) give us the probability lower bound of $\frac{3}{4}$. Also, the output of $A'(R, \ell', y, z)$ for such ℓ', z will trivially be (m, H).

To show the converse assume that there is a number m and a group H such that:

$$\begin{aligned} & \mathrm{Prob}_{y \in \{0,1\}^{2 \log^c \#R}}[(\exists \ell' z z' \in \{0,1\}^{3 \log^c \#R}), \text{ both } A'(R, \ell', y, z) \\ & \quad \text{and } B(R, \ell', y, z') \text{ accept and } A'(R, \ell', y, z) = (m, H)] \geq \frac{3}{4} \end{aligned} \tag{3.7}$$

Now if $(m, H) \neq (\#Aut(R), Aut(R))$ then the way A' outputs, it is clear that Merlin tried to "fool" Arthur and so by the Equations (3.5) we get that for some positive d':

$$\begin{aligned} & \mathrm{Prob}_{y \in \{0,1\}^{2 \log^c \#R}}[(\exists \ell' z z' \in \{0,1\}^{3 \log^c \#R}), \text{both } A'(R, \ell', y, z) \text{ and} \\ & B(R, \ell', y, z') \text{ accept } \mid A'(R, \ell', y, z) \neq (\#Aut(R), Aut(R))] \leq \frac{1}{2^{\log^{d'} \#R}} \end{aligned}$$

which together with the large probability lower bound of Equation (3.7) means that: $(m, H) = (\#Aut(R), Aut(R))$. Thus,

$$\begin{aligned} & \mathrm{Prob}_{y \in \{0,1\}^{2 \log^c \#R}}[(\exists \ell' z z' \in \{0,1\}^{3 \log^c \#R}), \text{ both } A'(R, \ell', y, z) \\ & \quad \text{and } B(R, \ell', y, z') \text{ accept and } A'(R, \ell', y, z) = (m, H)] \geq \frac{3}{4} \\ \Rightarrow \quad & f(R) = (m, H) \end{aligned} \tag{3.8}$$

Recall Equation (2.2) for the definition of fnAM, clearly, Equations (3.6) and (3.8) tell us that: $f \in$ fnAM. □

This completes the proof that GroupRA is in fnAM. The significance of this upper bound is that even though so many interesting and unclassified problems in NPreduce to GroupRA, it is not an NP-hard problem. Only a few naturally occuring examples of problems of intermediate complexity are known and this shows that GroupRA is one of them.

The next two sections study the problem of checking whether a given ring is rigid (i.e., has no nontrivial automorphism) and if not then finding a nontrivial automorphism. We will show that RA, the problem of determining if a given ring has a nontrivial automorphism, can be decided in deterministic polynomial time but finding a nontrivial automorphism (FRA) is as hard as integer factoring.

Thus, there appears to be a difference in the complexity of decision, search and counting versions of ring automorphism problems. Also, note the contrast that we (currently) have with the complexity of the corresponding versions for graph automorphism problems, for instance, GA is not known to be in P.

3.5 The Complexity of deciding the existence of a nontrivial automorphism.

Definition 3.5.1. A ring R is said to be rigid if it does not admit any nontrivial automorphism.

RA is then the computational problem of deciding if a given input ring is rigid. The aim of this section is to prove the following theorem.

Theorem 3.5.2. *$RA \in P$.*

We first derive a classification of finite rigid rings and then use that classification to devise an efficient algorithm for RA.

3.5.1 A classification of finite rigid rings.

In this subsection, we shall show here that those finite rings which do not have nontrivial automorphisms (rigid rings) have a nice mathematical description which will later be used to test rigidity in polynomial time. Indeed, we will show that

Theorem 3.5.3. *Let R be any finite ring with identity. R can be expressed as the direct sum of two rings*

$$R = R_{2pow} \otimes R_{odd},$$

where R_{2pow} is a power-of-2 sized ring while R_{odd} is an odd-sized ring. Then R is rigid if and only if

1. *R_{2pow} is of the form*

$$\begin{aligned} \mathbb{Z}/2^{\alpha_1}\mathbb{Z} \otimes \ldots \otimes \mathbb{Z}/2^{\alpha_n}\mathbb{Z} \text{ or} \\ (\mathbb{Z}/2\mathbb{Z})[x]/(x^2) \otimes \mathbb{Z}/2^{\alpha_1}\mathbb{Z} \otimes \ldots \otimes \mathbb{Z}/2^{\alpha_n}\mathbb{Z} \\ \text{where, } 1 \leq \alpha_1 < \alpha_2 < \ldots < \alpha_n. \end{aligned}$$

2. *R_{odd} is of the form*

$$\bigotimes_i \bigotimes_j \mathbb{Z}_{p_i^{\alpha_{ij}}} \text{ where, } p_i\text{'s are distinct odd primes and } 1 \leq \alpha_{i1} < \alpha_{i2} < \ldots.$$

Proof. It is easy to verify the following claim:

Claim 3.5.3.1. *A ring R is rigid if and only if each one of its indecomposable component rings is rigid and no two of these indecomposable components are isomorphic.*

This means that any arbitary rigid ring is just a direct sum of a set of non-isomorphic indecomposable rigid rings. Thus to get a classification of finite rigid rings, it is sufficient to get a classification of finite indecomposable rigid rings. In the rest of this proof we give such a requisite characterization of indecomposable rigid rings.

Let R be a ring given in basis form. Let us first dispose off the case when R is non-commutative.

Claim 3.5.3.2. *If R is a non-commutative ring then it has a nontrivial automorphism.*

Proof of Claim 3.5.3.2. It can be shown ([Len04]) that if the *units* in a ring R commute with the whole of R then R is generated by its units, and consequently R will be commutative. Thus, if R is a non-commutative ring then there is a *unit* $r \in R$ that doesn't commute with the whole of R. Then clearly the map $\phi : x \mapsto rxr^{-1}$ gives a nontrivial automorphism of R. □

When R is commutative we first consider the case of odd sized component subring R_{odd} of R.

Classification of R_{odd}. We will show that indecomposable *components* of a rigid commutative odd-sized ring R_{odd} are isomorphic to $\mathbb{Z}/p^m\mathbb{Z}$, for some odd prime p:

Claim 3.5.3.3. *If R_{odd} is an indecomposable rigid commutative odd-sized ring then $\exists$ prime p and $m \in \mathbb{N}$ such that, $R_{odd} \cong \mathbb{Z}/p^m\mathbb{Z}$.*

Proof of Claim 3.5.3.3. It is known (e.g. see [McD74]) that any indecomposable commutative ring R_{odd} contains an associated *Galois ring* G such that:

$$G = (\mathbb{Z}/p^m\mathbb{Z})[x]/(f(x))$$

where square-free $f(x)$ is irreducible over $\mathbb{Z}/p\mathbb{Z}$ and,

$$R_{odd} = G[x_1, \ldots, x_k]/(x_1^{n_1}, \ldots, x_k^{n_k}, g_1, \ldots, g_l)$$

where $x_1, \ldots, x_k$ form an irredundant generating set for R_{odd} over G and the g_i's are polynomials in $(x_1, \ldots, x_k)$.

Let $\mathcal{M} :=$ (ring generated by $x_1, \ldots, x_k$) be an ideal of R_{odd}, it will be nonzero if we assume $k \geq 1$. Let $t > 0$ be the least integer such that $\mathcal{M}^t = 0$.

Consider the case when $t > 2$. We can assume without loss of generality that x_1 cannot be expressed as a polynomial in $x_2, \ldots, x_k$ in the ring R_{odd}. Now choose an $\alpha \in \mathcal{M}^{t-1}$ such that no term in α is linear in x_1 and consider the map:

$$\phi : \begin{cases} x_1 & \mapsto x_1 + \alpha \\ x_2 & \mapsto x_2 \\ & \vdots \\ x_k & \mapsto x_k \end{cases}$$

ϕ is injective: otherwise a polynomial $h(x_1, ..., x_k)$ maps to 0, in R_{odd}, under ϕ. This means that $h(x_1 + \alpha, ..., x_k) = 0$ in R_{odd}. Now if $h(x_1, ..., x_k)$ had no linear occurrence of x_1 then $h(x_1 + \alpha, x_2, ..., x_k) = 0$ implies $h(x_1, ..., x_k) = 0$ (as $\alpha \cdot \mathcal{M} = 0$). On the other hand if $h(x_1, ..., x_k)$ has a linearly occurring x_1 then $h(x_1 + \alpha, x_2, ..., x_k) = 0$ implies that $x_1 =$ (an expression containing no linear term in x_1) which in turn means that x_1 can be expressed completely in terms of $x_2, ..., x_k$ which is a contradiction.

ϕ is onto: it is enough to show that in the ring R_{odd} we can obtain x_1 from $x_1 + \alpha, x_2, ..., x_k$. Since α is generated by $x_1, \ldots, x_l$ it can be expressed as a polynomial in $x_1, \ldots, x_l$. Viweing α as a polynomial in x_1. Let $\alpha = x_1 \cdot h(x_1, \ldots, x_k) + g(x_2, \ldots, x_k)$, where $h(x_1, ..., x_k)$ has no constant term. Then

$$\begin{aligned} x_1 + \alpha - g(x_2, ..., x_k) &= x_1 + x_1 \cdot h(x_1, ..., x_k) \\ &= x_1 + x_1 \cdot h(x_1 + \alpha, x_2, ..., x_k) \quad (\text{as } \alpha \cdot \mathcal{M} = 0) \\ &= x_1 \cdot (1 + h(x_1 + \alpha, x_2, ..., x_k)). \end{aligned}$$

Now $h(x_1 + \alpha, x_2, ..., x_k) \in \mathcal{M}$, and therefore by the property 2.1.5 of local rings, $(1 + h(x_1 + \alpha, x_2, ..., x_k))$ has to be invertible in R_{odd} and thus,

$$x_1 = [(x_1 + \alpha) - g(x_2, ..., x_k)] \cdot [1 + h(x_1 + \alpha, x_2, ..., x_k)]^{-1} \quad \text{in } R_{odd}.$$

Thus, ϕ induces a nontrivial automorphism of R_{odd}. This means that for R_{odd} to be rigid, we must have that the number of variables k is zero R is just a Galois ring. $R_{odd} = G$. If $f(x)$ is of degree > 1 then $(\mathbb{Z}/p\mathbb{Z})[x]/(f(x))$ has a nontrivial automorphism, the *Frobenius* automorphism sending $x \mapsto x^p$, which can be *Hensel lifted* (see [LN94]) to a nontrivial automorphism of $(\mathbb{Z}/p^m\mathbb{Z})[x]/(f(x))$ too. Thus, the only way that R_{odd} has no nontrivial automorphism is when degree of $f(x)$ is 1 meaning $R_{odd} = G = \mathbb{Z}/p^m\mathbb{Z}$.

Now suppose $t = 2$. If $k \geq 2$ then taking $\alpha = x_2$ in the above discussion gives us a nontrivial automorphism ϕ of R_{odd}. If $k = 1$ then the map $\phi : x_1 \mapsto 2x_1$ is a nontrivial automorphism of R_{odd}. If $k = 0$ then $R_{odd} = G$ and as shown before the only way that R_{odd} has no nontrivial automorphism is when $R_{odd} = G = \mathbb{Z}/p^m\mathbb{Z}$.

The last case of $t = 1$ means $\mathcal{M} = 0$ implying $R_{odd} = G$ which as before yields $R_{odd} = G = \mathbb{Z}/p^m\mathbb{Z}$. □

As a consequence of the above observations we have that any rigid commutative odd-sized ring R_{odd} looks like:

$$\bigotimes_i \bigotimes_j \mathbb{Z}_{p_i^{\alpha_{ij}}} \text{ where, } p_i\text{'s are distinct odd primes and } 1 \leq \alpha_{i1} < \alpha_{i2} < \ldots . \tag{3.9}$$

Classification of R_{2pow}. Let us now take up the case of the power-of-2 sized component subring R_{2pow} of R. We will show that R_{2pow} is rigid only if the indecomposable rings that appear in the decomposition of R_{2pow} are isomorphic to either $\mathbb{Z}/2^m\mathbb{Z}$ or $(\mathbb{Z}/2\mathbb{Z})[x]/(x^2)$.

Claim 3.5.3.4. *If R_{2pow} is an indecomposable rigid commutative power-of-2 sized ring then R_{2pow} is either $\mathbb{Z}/2^m\mathbb{Z}$ or $(\mathbb{Z}/2\mathbb{Z})[x]/(x^2)$.*

Proof of Claim 3.5.3.4. Recall the proof of the claim 3.5.3.3. The only case which needs to be handled in the case of *even* sized ring is when $t = 2$ and $k = 1$. The rigidity of R_{2pow} implies that the characteristic of R_{2pow} is 2 for otherwise $\phi : x_1 \mapsto 3x_1$ gives a nontrivial automorphism of R_{2pow}. Thus, *the* rigid ring with $t = 2, k = 1$ is $R = (\mathbb{Z}/2\mathbb{Z})[x_1]/(x_1^2)$. □

It follows from the above claim that a commutative power-of-2 sized ring is rigid iff it is isomorphic to one of the following:

$$\begin{gathered}\mathbb{Z}/2^{\alpha_1}\mathbb{Z} \otimes \ldots \otimes \mathbb{Z}/2^{\alpha_n}\mathbb{Z} \text{ or} \\ (\mathbb{Z}/2\mathbb{Z})[x]/(x^2) \otimes \mathbb{Z}/2^{\alpha_1}\mathbb{Z} \otimes \ldots \otimes \mathbb{Z}/2^{\alpha_n}\mathbb{Z} \qquad (3.10) \\ \text{where, } 1 \le \alpha_1 < \alpha_2 < \ldots < \alpha_n.\end{gathered}$$

Collecting these two classifications, we get the classification theorem 3.5.3 for finite rigid rings.

□

3.5.2 The Algorithm for RA

We now give the algorithm referred to in theorem 3.5.2 for testing the rigidity of a ring. Our algorithm for RA will test whether a given ring R is of the form given in the classification theorem (3.5.3) or not. As in the classification thereoem (3.5.3), suppose that the decomposition of a given input rings R is

$$R = R_{2pow} \otimes R_{odd}, \qquad (3.11)$$

where R_{2pow} is a power-of-2 sized ring and R_{odd} is an odd-sized ring. Note that since its easy to factor out powers of 2 from any integer, we can compute the decomposition of the additive group $(R, +)$ of R as the direct sum of two subgroups - one having power-of-2 size and another having odd size. This decomposition of $(R, +)$ then readily gives a decomposition of the form (3.11) of the input ring R. Note that now R is rigid if and only if both R_{2pow} and R_{odd} are rigid rings. In this way our problem boils down into cases - testing rigidity of R_{2pow} and that of R_{odd}.

Testing rigidity of R_{2pow}. Since we can factor polynomials over $\mathbb{Z}/2^m\mathbb{Z}$ we can compute the decomposition of R_{2pow} into indecomposable rings and check whether they are of the forms: $\mathbb{Z}/2^m\mathbb{Z}, (\mathbb{Z}/2\mathbb{Z})[x]/(x^2)$ or not. Hence, we can check the rigidity of power-of-2-sized rings in polynomial time.

Testing rigidity of R_{odd}. Let R_{odd} be given as:

$$(R_{odd}, +) = (\mathbb{Z}/m_1\mathbb{Z})e_1 \oplus \ldots \oplus (\mathbb{Z}/m_n\mathbb{Z})e_n$$

Here we can assume that $(m_1, \ldots, m_n) = (d_1^{\alpha_{11}}, d_1^{\alpha_{12}}, \ldots, d_2^{\alpha_{21}}, d_2^{\alpha_{22}}, \ldots, d_t^{\alpha_{t1}}, d_t^{\alpha_{t2}}, \ldots)$ where $d_1, \ldots, d_t$ are mutually coprime. For otherwise $\exists i \neq j$ s.t. $gcd(m_i, m_j) =: g > 1$ and can be used to break m_i or m_j into coprime factors $a, b \in \mathbb{Z}^{>1}$, hence, breaking $(R_{odd}, +)$ further by applying:

$$((\mathbb{Z}/ab\mathbb{Z})e_k, +) \cong (\mathbb{Z}/a\mathbb{Z})(be_k) \oplus (\mathbb{Z}/b\mathbb{Z})(ae_k)$$

We can repeatedly apply this process of refining the basis to get basis representations of the ring R_{odd} over:

$$\mathbb{Z}/d_1^{\alpha_{11}}\mathbb{Z} \oplus \mathbb{Z}/d_1^{\alpha_{12}}\mathbb{Z} \oplus \ldots \oplus \mathbb{Z}/d_2^{\alpha_{21}}\mathbb{Z} \oplus \ldots \oplus \mathbb{Z}/d_t^{\alpha_{t1}}\mathbb{Z} \oplus \ldots$$

$$\text{for some coprime } d_1, d_2, \ldots, d_t \in \mathbb{Z}^{>1}$$

Let us define for all $1 \leq i \leq t$,

$$R_i := \{r \in R_{odd} \mid r \text{ has a power-of-}d_i \text{ additive order}\}$$

Now since the d_i's are mutually coprime $R_{odd} \cong \bigotimes_{i=1}^{t} R_i$ (as in the proof of proposition 3.4.2). Thus, R_{odd} has a nontrivial automorphism iff $\exists i \in [t]$, R_i has a nontrivial automorphism. Consequently, we can assume without loss of generality that the additive basis of the rings R_{odd} is given in the form:

$$(R_{odd}, +) = (\mathbb{Z}/d^{\alpha_1}\mathbb{Z})e_1 \oplus \ldots \oplus (\mathbb{Z}/d^{\alpha_n}\mathbb{Z})e_n \tag{3.12}$$

We can also assume that α_i's are distinct (say, $1 \leq \alpha_1 < \alpha_2 < \ldots < \alpha_n$) otherwise R_{odd} would not be rigid as it would not be of the form in the classification theorem (3.5.3).

Thus we need to check if a given ring R_{odd} is of the form:

$$\mathbb{Z}/d^{\alpha_1}\mathbb{Z} \otimes \ldots \otimes \mathbb{Z}/d^{\alpha_n}\mathbb{Z} \tag{3.13}$$

Remark. There do exist rings whose additive group is of the form (3.12) but the rings themselves are not of the form (3.13). For example, the ring $R \overset{\text{def}}{=} (\mathbb{Z}/d^2\mathbb{Z})[x]/\langle x^2, dx\rangle$ has additive group isomorphic to $\mathbb{Z}/d^2\mathbb{Z} \oplus \mathbb{Z}/d\mathbb{Z}$ but R is *not* isomorphic to $\mathbb{Z}/d^2\mathbb{Z} \otimes \mathbb{Z}/d\mathbb{Z}$.

Overview of the algorithm. Now we sketch an algorithm to check whether R_{odd} is isomorphic to:

$$\mathbb{Z}/d^{\alpha_1}\mathbb{Z} \otimes \ldots \otimes \mathbb{Z}/d^{\alpha_n}\mathbb{Z} \quad \text{for } \alpha_1 < \ldots < \alpha_n.$$

Our algorithm proceeds by decomposing R_{odd} into $\mathbb{Z}/d^{\alpha_1}\mathbb{Z} \otimes R'$ and then recursively verifying that the component ring R' is of the form

$$\mathbb{Z}/d^{\alpha_2}\mathbb{Z} \otimes \ldots \otimes \mathbb{Z}/d^{\alpha_n}\mathbb{Z} \quad \text{for } \alpha_1 < \alpha_2 \ldots < \alpha_n.$$

The key observation behind obtaining the decomposition of R_{odd} into $\mathbb{Z}/d^{\alpha_1}\mathbb{Z} \otimes R'$ is the following claim which is easy to verify:

Claim 3.5.3.5. *If*

$$\psi : R_{odd} \mapsto \mathbb{Z}/d^{\alpha_1}\mathbb{Z} \otimes \ldots \otimes \mathbb{Z}/d^{\alpha_n}\mathbb{Z}$$

is an isomorphism and

$$(R_{odd}, +) = (\mathbb{Z}/d^{\alpha_1}\mathbb{Z})e_1 \oplus \ldots \oplus (\mathbb{Z}/d^{\alpha_n}\mathbb{Z})e_n$$

then $\psi(e_1) = (\beta_1, \beta_2, \ldots, \beta_n)$ *where* $\beta_1 \in (\mathbb{Z}/d\mathbb{Z})^*$ *and* $d|\beta_2, \ldots, \beta_n$*, so that if* $f(x) \in \mathbb{Z}[x]$ *is the minimal polynomial of* e_1 *in* R_{odd} *then*

$$f(x)\ (mod\ d) = x^l \cdot (x - (\beta_1\ (mod\ d))) \quad for \ \ some\ l \in \mathbb{Z}^{\geq 0}.$$

Following the above claim, we compute $\beta_1 \in \mathbb{Z}/d^{\alpha_1}\mathbb{Z}$ and thereby obtain the zero divisor $(e_1 - \beta_1)$ of R_{odd} and this zero divisor is then used in the standard way to decompose R_{odd}.

Algorithm: Determine if R_{odd} is of the form (3.13).

S-1. Compute $f(x) :=$ minpoly of e_1 over $\mathbb{Z}/d^{\alpha_n}\mathbb{Z}$. This can be found out by checking whether e_1^i can be written as a linear combination of $1, e_1, \ldots, e_1^{i-1}$ which amounts to doing linear algebra (mod d^{α_n}).

S-2. If $R_{odd} \cong \mathbb{Z}/d^{\alpha_1}\mathbb{Z} \otimes \ldots \otimes \mathbb{Z}/d^{\alpha_n}\mathbb{Z}$ then say $e_1 = (\beta_1, \ldots, \beta_n)$ where $\beta_i \in \mathbb{Z}/d^{\alpha_i}\mathbb{Z}$. Also, since e_1 has characteristic d^{α_1} and $\alpha_1 \lneq \alpha_2, \ldots, \alpha_n$ we can deduce: β_1 is coprime to d and $d|\beta_2, \ldots, \beta_n$.

These observations mean that:

$$\begin{aligned} f(x) &= \text{lcm}_{i=1}^{n} \{\text{minpoly of } \beta_i \text{ over } \mathbb{Z}/d^{\alpha_i}\mathbb{Z}\} \\ &\equiv (x - \beta_1)x^l \ (\text{mod } d), \quad \text{for some } l \in \mathbb{Z}^{\geq 0} \end{aligned}$$

or else R_{odd} is not of the form (3.13). So we have a non-repeating root $\beta_1(\text{mod } d)$ of $f(x)(\text{mod } d)$ and we can use Hensel lifting (see section 2.1.7) to find a root of $f(x)(\text{mod } d^{\alpha_1})$, which gives $\beta_1(\text{mod } d^{\alpha_1})$.

S-3. Consider $e_1 - \beta_1 = (0, \beta_2 - \beta_1, \ldots, \beta_n - \beta_1)$. Note that $\beta_2 - \beta_1, \ldots, \beta_n - \beta_1$ are all coprime to d. So if we compute (using linear algebra)

$$R_1 := \{\gamma \in R_{odd} \mid (e_1 - \beta_1)\gamma = 0\}$$

then $R_1 \cong \mathbb{Z}/d^{\alpha_1}\mathbb{Z}$ or else R_{odd} is not of the form (3.13).

S-4. Let $\hat{e}_1 \in R_{odd}$ be the unity of R_1. Compute $R_1^{\perp} := \{\gamma \in R_{odd} \mid \hat{e}_1\gamma = 0\}$. Check that $R_{odd} = R_1 \otimes R_1^{\perp}$ otherwise R_{odd} is not of the form (3.13).

S-5. Recursively check whether $R_1^{\perp} \cong \mathbb{Z}/d^{\alpha_2}\mathbb{Z} \otimes \ldots \otimes \mathbb{Z}/d^{\alpha_n}\mathbb{Z}$ or not.

Remark. Note that our algorithm for ring automorphism does not imply an efficient algorithm for the Graph Automorphism problem because the construction of the ring associated with a graph G (as in section (3.3.4)) gives us a local $\mathbb{F}_5$-algebra $R(G)$ which has lots of nontrivial automorphism already as in the proof of lemma (3.5.3.3).

3.6 Computing a nontrivial automorphism.

We just saw that deciding whether a ring has a nontrivial automorphism is in P. Here we give evidence that the search version of this problem is apparently harder. We show that FRA is as hard as integer factoring (IF).

Theorem 3.6.1. *IF* $\equiv_T^{ZPP}$ *FRA.*

Proof. Let us first see how we can find a nontrivial ring automorphism if we can do integer factoring. Suppose the given ring R is non-commutative then we know from the proof of claim 3.5.3.2: there is a *unit* of R that does not commute with the whole of R and thus defines a nontrivial automorphism. So we compute the multiplicative generators of R^* in *randomized* polynomial time and surely one of the generators will not commute with the whole of ring R.

Now assume the given ring R is commutative. It can be decomposed into local rings, as remarked after lemma 3.3.4, in expected polynomial time using randomized methods for polynomial factorization and oracle of integer factorization. Once we have local rings we can output nontrivial automorphisms like ϕ in the proof of claim 3.5.3.3.

Conversely, suppose we can find nontrivial automorphisms of rings and n is a given number. We can assume that n has no small($\leq (\log n)^3$) prime factor p for clearly we can find such small prime factors in polynomial-time. Let $n = p^a \cdot m$ where p^a is the highest power of the prime p which divides n and m is coprime to p. Randomly choose a monic cubic polynomial $f(x) \in (\mathbb{Z}/n\mathbb{Z})[x]$. Define $R := (\mathbb{Z}/n\mathbb{Z})[x]/(f(x))$ and suppose we can find a nontrivial automorphism ϕ of R. It follows from the distribution of irreducible polynomials over finite fields ([LN94]) that with probability $\sim \frac{1}{9}$, $f \pmod n$ satisfies the following properties:

- $f(\bmod\ n)$ is squarefree. Equivalently, n is coprime to the discriminant, Δ_f, of f.
- $f(\bmod\ m)$ is irreducible. That is, there exists a prime $q|m$ such that $f(\bmod\ q)$ is irreducible.
- $f(\bmod\ p)$ has exactly two irreducible factors f_1, f_2, say f_1 is linear.

Thus,

$$R \cong (\mathbb{Z}/p^a\mathbb{Z}) \otimes (\mathbb{Z}/p^a\mathbb{Z})[x]/(f_2(x)) \otimes (\mathbb{Z}/m\mathbb{Z})[x]/(f(x)).$$

Note that we can compute R^ϕ, the set of elements of R fixed by ϕ, using linear algebra (if at any point we cannot invert an element (mod n), we get a factor of n). As ϕ is a nontrivial automorphism of R we have that ϕ is identity on atmost one of the component rings $(\mathbb{Z}/p^a\mathbb{Z})[x]/(f_2(x))$ or $(\mathbb{Z}/m\mathbb{Z})[x]/(f(x))$. Thus, we have three cases:

C1). If ϕ fixes $(\mathbb{Z}/p^a\mathbb{Z})[x]/(f_2(x))$:
Then $R^\phi \cong (\mathbb{Z}/p^a\mathbb{Z}) \otimes (\mathbb{Z}/p^a\mathbb{Z})[x]/(f_2(x)) \otimes (\mathbb{Z}/m\mathbb{Z}/(f(x)))^\phi$. Thus, $|R^\phi| = p^{3a}m_1$ where $m_1 \neq m^3$ as ϕ moves $(\mathbb{Z}/m\mathbb{Z}/(f(x)))$.

C2). If ϕ fixes $(\mathbb{Z}/m\mathbb{Z})[x]/(f(x))$:
Then $R^\phi \cong (\mathbb{Z}/p^a\mathbb{Z}) \otimes (\mathbb{Z}/p^a\mathbb{Z}) \otimes (\mathbb{Z}/m\mathbb{Z})[x]/(f(x))$. Thus, $|R^\phi| = p^{2a}m^3$.

C3). If ϕ moves both $(\mathbb{Z}/p^a\mathbb{Z})[x]/(f_2(x))$ and $(\mathbb{Z}/m\mathbb{Z})[x]/(f(x))$:
Then $R^\phi \cong (\mathbb{Z}/p^a\mathbb{Z}) \otimes (\mathbb{Z}/p^a\mathbb{Z}) \otimes (\mathbb{Z}/m\mathbb{Z})[x]/(f(x))^\phi$. Thus, $|R^\phi| = p^{2a} \cdot m_1$, where $m_1 \neq m^2$ because $f(\bmod\ m)$ is irreducible. (if $q^b|m$ be such that $f(\bmod\ q)$ is irreducible, then $(\mathbb{Z}/q^b\mathbb{Z})[x]/(f(x))^\phi$ has size precisely q^b.)

Since, the size of R^{ϕ} is in no case of the form n, n^2 or n^3, the process of finding R^{ϕ} by doing linear algebra (mod n) is going to yield a factor of n. In particular, this means that if the matrix describing ϕ over the natural additive basis $\{1, x, x^2\}$ is:

$$A := \begin{pmatrix} 1 & 0 & 0 \\ a_0 & a_1 & a_2 \\ b_0 & b_1 & b_2 \end{pmatrix}$$

then the determinant of one of the submatrices of $(A - I)$ will have a nontrivial gcd with n.

This idea can be extended to the case of composite n having more prime factors.

Thus, the two problems: finding nontrivial automorphisms of commutative rings and integer factoring have the same complexity (with respect to randomized polynomial time reductions). □

3.7 Discussion

In this chapter, we studied the complexity of computing the automorphism group of a given ring and its variants. A much finer classification is given in [KS05, Sax06]. Some related open problems and conjectures are given in the chapter on open problems (chapter 8).

Chapter 4

Polynomial Identity Testing for Depth-3 Cicuits

Summary: In this chaper, we study the identity testing problem for depth 3 arithmetic circuits ($\Sigma\Pi\Sigma$ circuits). We give the first deterministic polynomial time identity test for $\Sigma\Pi\Sigma$ circuits with bounded top fanin.

4.1 Introduction

Polynomial Identity Testing (PIT) is the following problem: given an arithmetic circuit $\mathcal{C}$ computing a polynomial $p(x_1, x_2, \cdots, x_n)$ over a field $\mathbb{F}$, determine if the polynomial is identically zero. Besides being an interesting problem in itself, many other well-known problems such as Primality Testing and Bipartite Matching also reduce to PIT. Moreover fundamental structural results in complexity theory such as IP=PSPACE and the PCP theorem involve the use of identity testing.

The first randomized algorithm for identity testing was discovered independently by Schwartz [Sch80] and Zippel [Zip79] and it involves evaluating the polynomial at a random point and accepting if and only if the polynomial evaluates to zero at that point. This was followed by randomized algorithms that used fewer random bits [CK00, LV98, AB03] and a derandomization of the polynomial involved in primality testing [AKS04] but a complete derandomization remains distant.

A surprising observation was made by Impaggliazzo and Kabanets [KI04] who showed that efficient deterministic algorithms for identity testing would also imply strong arithmetic circuit lower bounds. More specifically, they showed that if identity testing has an efficient deterministic polynomial time algorithm then NEXP does not have polynomial size *arithmetic* circuits. This result gave further impetus to research on this problem and subsequently algorithms were developed for some restricted models of arithmetic circuits.

Raz and Shpilka [RS04] gave a deterministic polynomial time algorithm for non-commutative formulas. Klivans and Spielman [KS01] noted that even for depth 3 circuits where the fanin of the topmost gate was bounded, deterministic identity testing was an open problem. Subsequently, Dvir and Shpilka [DS05] gave a deterministic *quasipolynomial time* algorithm for depth 3 arithmetic circuits ($\Sigma\Pi\Sigma$ circuits) where the fanin of the topmost gate is bounded (note that if the topmost gate is a Π gate than the polynomial is zero if and only if one of the factors is zero and the problem is then easily solved).

Example: The circuit

$$\mathcal{C}(y, x_1, x_2) \overset{\text{def}}{=} (y) \cdot (y + x_1 + x_2) + (x_1) \cdot (x_2) - (y + x_1) \cdot (y + x_2)$$

is a $\Sigma\Pi\Sigma$-circuit computing the identically zero polynomial over the field $\mathbb{Q}$ of rational numbers.

In this paper, we give a *deterministic polynomial time* algorithm for the identity testing of such $\Sigma\Pi\Sigma$ circuits. Our main theorem is:

Theorem 4.1.1. *There exists a deterministic algorithm that on input a circuit $\mathcal{C}$ of depth 3 and degree d over a field $\mathbb{F}$, determines if the polynomial computed by the circuit is identically zero in time poly$(n \cdot d^k)$, where k is the fanin of the topmost addition gate and n is the number of inputs. In particular if k is bounded, then we get a deterministic polynomial time algorithm for identity testing of depth 3 circuits.*

The rest of this chapter is organized as follows. Section 4.2 gives an overview of $\Sigma\Pi\Sigma$ circuits. Then section 4.3 proves a generalization of the well-known Chinese Remaindering Theorem which is crucial to our algorithm. Finally, section 4.4 describes the identity test for $\Sigma\Pi\Sigma$ circuits of bounded top fanin.

4.2 $\Sigma\Pi\Sigma$ Arithmetic Circuits

As noted by Impagliazzo and Kabanets, the Polynomial Identity Testing problem is closely related to proving arithmetic circuit lower bounds. Proving lower bounds for general arithmetic circuits is one of the central problems of complexity theory. Due to the difficulty of the problem research has focused on restricted models like monotone circuits and bounded depth circuits. Only weak lower bounds are known for bounded depth arithmetic circuits [Pud94, RS01]. Thus, a more restricted model was considered – the model of depth 3 arithmetic circuits (also called $\Sigma\Pi\Sigma$ circuits if we assume alternate addition and multipication gates with addition gate at the top). A $\Sigma\Pi\Sigma$ circuit computes a polynomial of the form:

$$\mathcal{C}(\overline{x}) = \sum_{i=1}^{k} \beta_i \prod_{j=1}^{d_i} L_{ij}(\overline{x}) \tag{4.1}$$

where L_{ij}'s are homogeneous linear functions (or linear forms). Exponential lower bounds on the size of $\Sigma\Pi\Sigma$ arithmetic circuits has been shown over *fixed size* finite fields [GK98]. For general $\Sigma\Pi\Sigma$ circuits over large/infinite fields only the quadratic lower bound of [SW99] is known.

No efficient algorithm for identity testing of $\Sigma\Pi\Sigma$ circuits is known. Here we are interested in studying the identity testing problem for a restricted case of $\Sigma\Pi\Sigma$ circuits – when the top fanin, k, is bounded. This case was posed as a challenge by Klivans and Spielman [KS01] and a *quasipolynomial time* algorithm was given by Dvir and Shpilka [DS05].

4.2.1 Previous Approaches

Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma$ circuit, as in Equation (4.1), computing the zero polynomial. We will call $\mathcal{C}$ to be *minimal* if no proper subset of the multiplication gates of $\mathcal{C}$ sums to zero. We say that $\mathcal{C}$ is *simple* if there is no linear function that appears in all the multiplication gates (up to a multiplicative constant). *Rank* of $\mathcal{C}$ is the rank of the linear forms appearing in $\mathcal{C}$.

The quasipolynomial time algorithm of [DS05] is based on the following result – rank of a minimal and simple $\Sigma\Pi\Sigma$ circuit with bounded top fanin and computing zero is “small”. Formally, the result says:

Theorem 4.2.1. (Thm 1.4 of [DS05]). Let $k \geq 3, d \geq 2$, and let $\mathcal{C} \equiv 0$ be a simple and minimal $\Sigma\Pi\Sigma$ circuit of degree d with k multiplication gates and n inputs, then $\mathrm{rank}(\mathcal{C}) \leq 2^{O(k^2)} \log(d)^{k-2}$.

Effectively, this means that if we have such a circuit $\mathcal{C}$ and k is a constant then we can check whether it is zero or not in time $O(d^{\mathrm{rank}(\mathcal{C})}) = 2^{O(\log(d)^{k-1})}$. This gave hope of finding a polynomial time algorithm if we can improve the upper bound on the rank($\mathcal{C}$) to a constant (i.e. independent of d). Infact, [DS05] conjectured that rank($\mathcal{C}$) = $O(k)$. Unfortunately this approach is unlikely to work at least over finite fields of fixed characteristic as shown by Nitin Saxena in [Sax06]. [Sax06] gives the following identity over finite fields of *fixed* characteristic p that contradicts this conjecture.

Lemma 4.2.2. *(***Saxena***, [Sax06]). Let p be an odd prime. Define:*

$$\mathcal{C}(x_1, \ldots, x_m) \overset{\text{def}}{=} \sum_{i=0}^{p-1} \prod_{\substack{b_1,\ldots,b_m \in \mathbb{F}_p \\ b_1+\cdots+b_m \equiv i (mod\ p)}} (b_1x_1 + \cdots + b_mx_m)$$

Then, over $\mathbb{F}_p$, $\mathcal{C}$ is a simple and minimal $\Sigma\Pi\Sigma$ zero circuit of degree $d = p^{m-1}$ with $k = p$ multiplication gates and having "unbounded" rank($\mathcal{C}$) $= \log_p(d) + 1$.

Thus, methods of [DS05] are unlikely to give an efficient algorithm (at least over all fields) and we give new techniques in section 3 that solve the problem.

4.2.2 Our Approach

We now give the basic idea behind our approach to this problem after introducing a little bit of notation.

▮ *Terminology - Leading monomial and leading coefficient.*

Let $\mathbb{F}$ be a field and $\succeq$ be the graded-lexicographic ordering on monomials in $\mathbb{F}[x_1, \ldots, x_n]$. That is, $\succeq$ ranks monomials by their total degree and breaks ties by using lexicographic ordering. For $f(\bar{\mathbf{x}}) \in \mathbb{F}[\bar{\mathbf{x}}]$:

- The *leading monomial* of $f(\bar{\mathbf{x}})$, written $LM(f(\bar{\mathbf{x}}))$, is the monomial which is ranked highest under $\succeq$ of all monomials which have nonzero coefficients in $f(\bar{\mathbf{x}})$.
- For a monomial $x_1^{\alpha_1}x_2^{\alpha_2}\ldots x_n^{\alpha_n}$ and a polynomial $f(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$, we will denote by $\mathrm{Coeff}(x_1^{\alpha_1}x_2^{\alpha_2}\ldots x_n^{\alpha_n}, f(x_1, \ldots, x_n))$ the coefficient of the monomial $x_1^{\alpha_1}x_2^{\alpha_2}\ldots x_n^{\alpha_n}$ in $f(x_1, \ldots, x_n)$.

Note that the leading monomial satisfies the following property:

Property. For $f_1(\bar{\mathbf{x}}), f_2(\bar{\mathbf{x}}) \in \mathbb{F}[\bar{\mathbf{x}}]$,

(i) $LM(f_1(\bar{\mathbf{x}}) \cdot f_2(\bar{\mathbf{x}})) = LM(f_1(\bar{\mathbf{x}})) \cdot LM(f_2(\bar{\mathbf{x}}))$. In particular, if $f_1(\bar{\mathbf{x}})$ divides $f_2(\bar{\mathbf{x}})$ then $LM(f_1(\bar{\mathbf{x}}))$ divides $LM(f_2(\bar{\mathbf{x}}))$.

(ii) If $LM(f_1(\bar{\mathbf{x}})) \succeq LM(f_2(\bar{\mathbf{x}}))$ then $LM(f_1(\bar{\mathbf{x}})) \succeq LM(f_1(\bar{\mathbf{x}}) + f_2(\bar{\mathbf{x}}))$.

▮ *The Idea*

The input is a circuit $\mathcal{C}(x_1, \ldots, x_n)$ in $\mathbb{F}[x_1, \ldots, x_n]$ which looks like:

$$\mathcal{C} = \beta_1 \cdot T_1 + \beta_2 \cdot T_2 + \ldots + \beta_k \cdot T_k$$

where each T_i is a product of linear forms

$$T_i = L_{i1} \cdot L_{i2} \cdot \ldots \cdot L_{id}$$

and where each L_{ij} is a linear form:

$$L_{ij} = a_{ij1}x_1 + a_{ij2}x_2 + \cdots + a_{ijn}x_n$$

for some $a_{ij1}, a_{ij2}, \cdots, a_{ijn} \in \mathbb{F}$. We want to check if $\mathcal{C}$ computes the identically zero polynomial over $\mathbb{F}$. By rearranging the terms if necessary we can assume without loss of generality that

$$LM(T_1) \succeq LM(T_2) \succeq \ldots \succeq LM(T_k).$$

Then by property (ii) of leading monomials we have

$$LM(T_1) \succeq LM(\mathcal{C}). \tag{4.2}$$

We first verify that T_1 divides $\mathcal{C}$ in a recursive manner to be described a short while later. We next check if Coeff$(LM(T_1), \mathcal{C}(x_1, \ldots, x_n)) = 0$. We accept $\mathcal{C}$ if and only if it passes both the tests. Clearly, if $\mathcal{C}(x_1, \ldots, x_n) = 0$, the input will pass both the tests and our algorithm will correctly identify $\mathcal{C}(x_1, \ldots, x_n)$ as the zero polynomial. So we will assume that $\mathcal{C}(x_1, \ldots, x_n) \neq 0$. In that case, by Property (i) of LM we then have

$$LM(\mathcal{C}) \succeq LM(T_1). \tag{4.3}$$

Combining equations (4.2) and (4.3), we get

$$LM(\mathcal{C}) = LM(T_1).$$

But

$$\text{Coeff}(LM(T_1), \mathcal{C}(x_1, \ldots, x_n)) = 0,$$

implying that

$$\text{Coeff}(LM(\mathcal{C}), \mathcal{C}(x_1, \ldots, x_n)) = 0,$$

a contradiction since $\mathcal{C}(x_1, \ldots, x_n)$ was assumed to be non-zero.

Checking that T_1 divides $\mathcal{C}(x_1, \ldots, x_n)$. We have $T_1 = L_{11} \cdot L_{12} \cdot \ldots \cdot L_{1d}$. We recursively verify that $\mathcal{C} \equiv 0 \pmod{L_{1j}}$ for all $1 \leq j \leq d$. Note that T_1 vanishes modulo L_{1j} and that $\mathbb{F}[x_1, \ldots, x_n]/\langle L_{1j} \rangle \cong \mathbb{F}[y_1, \ldots, y_{n-1}]$ is isomorphic to a polynomial ring in $(n-1)$ variables over the field $\mathbb{F}$. Therefore verifying $\mathcal{C} \equiv 0 \pmod{L_{1j}}$ amounts to identity testing of a $\Sigma\Pi\Sigma$ circuit of top fanin $(k-1)$ in $(n-1)$ variables over the fied $\mathbb{F}$.

Having verified that $\mathcal{C} \equiv 0 \pmod{L_{1j}}$ for all $1 \leq j \leq d$, we can deduce by the Chinese Remaindering Theorem that $L \overset{\text{def}}{=} \textsc{Lcm}(L_{11}, L_{12}, \ldots, L_{1d})$ divides $\mathcal{C}$. Now if $L = T_1$ then we are done.

In general, however there would exist pathological cases in which T_1 has repeated factors and L properly divides T_1. The algorithm for the general case has the same structure as above, except that we now work with polynomials over local rings instead of fields. Our main tool will be a generalization of the Chinese Remainder Theorem (CRT). The next section is devoted to this generalization of CRT.

4.3 Chinese remaindering

In our algorithm, the polynomials that we get will be over some *local* ring $R \supset \mathbb{F}$ instead of being over $\mathbb{F}$ but we can show that the chinese remaindering property of polynomials in $\mathbb{F}[z_1, \ldots, z_n]$ continues to hold in $R[z_1, \ldots, z_n]$. Specifically, we need that:

> **Chinese Remaindering Theorem**: If *'coprime'* $f(z_1, \ldots, z_n)$, $g(z_1, \ldots, z_n)$ divide $p(z_1, \ldots, z_n)$ then $f \cdot g \mid p$ over R.

4.3.1 Notation and Terminology.

▮ *Terminology - Natural Ring Homomorphism.*

Let R be a local ring over a field $\mathbb{F}$ with maximal ideal $\mathcal{M}$. Then every element $r \in R$ can be written uniquely as $r = \alpha + m$ where $\alpha \in \mathbb{F}$ and $m \in \mathcal{M}$ is a nilpotent element of R. By the term *natural ring homomorphism from R to $\mathbb{F}$*, we will mean the unique homomorphism

$\phi : R \mapsto \mathbb{F}$ that maps every element in $\mathcal{M}$ to zero in $\mathbb{F}$. That is, $\phi(r) = \alpha$. The map ϕ then extends in a natural way to a homomorphism from the polynomial ring $R[z_1, \ldots, z_n]$ to the polynomial ring $\mathbb{F}[z_1, \ldots, z_n]$ so that the polynomial $\sum_\alpha a_\alpha \bar{\mathbf{z}}^\alpha$ is mapped to the polynomial $\sum_\alpha \phi(a_\alpha)\bar{\mathbf{z}}^\alpha$. We will say that two polynomials $f(\bar{\mathbf{z}})$ and $g(\bar{\mathbf{z}})$ in $R[\bar{\mathbf{z}}]$ are *coprime* if and only if the corresponding polynomials $\phi(f(\bar{\mathbf{z}}))$ and $\phi(g(\bar{\mathbf{z}}))$ are coprime.

▮ *Notation - Set of Linear Forms over R.*

Let R be a local ring over a field $\mathbb{F}$ with maximal ideal $\mathcal{M}$. We will denote by $LF_{R/\mathbb{F}}(\bar{\mathbf{x}})$ the set of all linear forms in n variables $\bar{\mathbf{x}} = (x_1, x_2, \ldots, x_n)$ over R with coefficients from $\mathbb{F}$. That is,

$$LF_{R/\mathbb{F}}(x_1, \ldots, x_n) = \{\sum_{i=1}^{i=n} a_i x_i + m | m \in \mathcal{M}, \;\; a_i \in \mathbb{F} \;, \;\; \exists i : a_i \neq 0\}$$

4.3.2 Preliminaries

For any ring S, we can define the *ring of fractions* S^{fr} of a ring S as the set of elements $\frac{u}{v}$ where, $u, v \in S$ and v is not a zero divisor of S. Clearly, S^{fr} is also a ring. We will be considering polynomials over rings S and S^{fr}. A polynomial $g(z) \in S[z]$ is called *monic* if its leading coefficient is a unit of S. The following is a well known lemma that relates polynomial factorization over the ring S to its ring of fractions S^{fr}.

Lemma 4.3.1 (Gauss' Lemma)**.** *Suppose $f(z), g(z) \in S[z]$ and $h(z) \in S^{fr}[z]$ such that: $f(z) = g(z)h(z)$. If $g(z)$ is monic then $h(z) \in S[z]$.*

Proof. Let the degrees of f, g and h be α, β and γ respectively. Let

$$f(z) = \sum_{i=0}^{\alpha} f_i z^i \;\text{ where }\; f_i \in S \;\;,$$

$$g(z) = \sum_{i=0}^{\beta} g_i z^i \;\text{ where }\; g_\beta = 1, g_i \in S \;\text{ and}$$

$$h(z) = \sum_{i=0}^{i=\gamma} h_i z^i \;\text{ where }\; h_i \in S^{\text{fr}}.$$

Suppose if possible that $h(z) \notin S[z]$. Let $k \in [1 \cdots \gamma]$ be the largest integer such that h_k, the coefficient of z^k in $h(z)$, does not belong to S. Now the coefficient of $z^{\beta+k}$ in $g(z)h(z)$

is

$$f_{\beta+k} = h_k g_\beta + \sum_{j=1}^{\beta} g_{\beta-j} h_{k+j}$$

Thus $f_{\beta+k}$ belongs to S^{fr} but not to S. This is a contradiction for $f(z) \in S[z]$.

□

4.3.3 Properties of multivariate polynomials over local rings

In this section we will show that (multivariate) polynomials over local rings have divisibility properties analogous to those of polynomials over fields. In showing this, we will often make use of linear transformation of variables. We start out with a lemma which shows that after the application of suitable linear transformation, any polynomial $p(x_1, x_2, \cdots, x_n) \in \mathbb{F}[x_1, x_2, \cdots, x_n]$ of total degree d can be transformed into a polynomial $\hat{p}$ that is monic of degree d with respect to the variable x_1.

Lemma 4.3.2. *Let $\mathbb{F}$ be a field of size at least d and $p(x_1, x_2, \cdots, x_n) \in \mathbb{F}[x_1, x_2, \cdots, x_n]$ be any polynomial of total degree d. Then there exists a linear transformation ϕ , $\phi : x_i \mapsto \sum_{j=1}^{n} \alpha_{ij} x_j$ such that $\hat{p}(x_1, x_2, \cdots, x_n) \overset{\mathrm{def}}{=} p(\phi(x_1), \phi(x_2), \cdots, \phi(x_n))$ is monic of degree d with respect to the variable x_1. Indeed a random linear transformation ϕ will work with high probability.*

Proof. Let $p(\bar{\mathbf{x}}) = q(\bar{\mathbf{x}}) + r(\bar{\mathbf{x}})$ where $q(\bar{\mathbf{x}}) \neq 0$ is a homogeneous polynomial of degree d and $r(\bar{\mathbf{x}})$ consists of all the remaining smaller degree terms of $p(\bar{\mathbf{x}})$. Then the coefficient of x_1^d in $\hat{p}(\bar{\mathbf{x}})$ is simply $q(\alpha_{11}, \alpha_{21}, \cdots, \alpha_{n1})$.

By the Schwartz-Zippel lemma [Sch79], $q(\alpha_{11}, \alpha_{21}, \cdots, \alpha_{n1}) \neq 0$ with high probability and thus $\hat{p}(\bar{\mathbf{x}})$ is monic in x_1 with high probability.

□

The same proof can now be extended to local rings over a field.

Corollary 4.3.3. *Let $\mathbb{F}$ be a field of size at least d and R a local ring over $\mathbb{F}$. Let $p(x_1, x_2, \cdots, x_n) \in R[x_1, x_2, \cdots, x_n]$ be any polynomial of total degree d. Then there exists a linear transformation ϕ , $\phi : x_i \mapsto \sum_{j=1}^{n} \alpha_{ij} x_j$ such that $\hat{p}(x_1, x_2, \cdots, x_n) \overset{\mathrm{def}}{=} p(\phi(x_1), \phi(x_2), \cdots, \phi(x_n))$ is monic of degree d with respect to the variable x_1. Indeed a random linear transformation ϕ will work with high probability.*

Throughout the rest of this section we will assume that R is a local ring over a field $\mathbb{F}$ and the natural ring homomorphism from R to $\mathbb{F}$ is ϕ. The natural extension of the map ϕ to a homomorphism from $R[z_1, z_2, \cdots, z_n]$ to $\mathbb{F}[z_1, z_2, \cdots, z_n]$ will also be denoted by ϕ. The unique maximal ideal of R is $\mathcal{M}$ and t is the least integer such that $\mathcal{M}^t = 0$ in R.

Lemma 4.3.4. *Let R be a local ring and $p, f, g \in R[z_1, z_2, \cdots z_n]$ be multivariate polynomials such that $\phi(f)$ and $\phi(g)$ are coprime.*

$$\begin{aligned} \textit{If} \quad & p \equiv 0 \ \ (\textit{mod } f) \\ \textit{and} \quad & p \equiv 0 \ \ (\textit{mod } g) \\ \textit{then} \quad & p \equiv 0 \ \ (\textit{mod } f \cdot g). \end{aligned}$$

Proof. Let the (total) degrees of $\phi(f)$ and $\phi(g)$ be d_f and d_g respectively. By corollary 4.3.3 we can apply a suitable invertible linear transformation on the variables $z_1, z_2, \cdots, z_n$, if needed, and can thus assume without loss of generality that the coefficients of $z_n^{d_f}$ in f and that of $z_n^{d_g}$ in g are both units of R. Consequently, in the product fg the coefficient of $z_n^{d_f+d_g}$ is also a unit.

Now think of f and g as polynomials in one variable z_n with coefficients coming from the ring of fractions, $R(z_1, z_2, \cdots, z_{n-1})$, of $R[z_1, z_2, \cdots, z_{n-1}]$. Now since $\phi(f)$ and $\phi(g)$ are coprime over $\mathbb{F}$, they are also coprime as univariate polynomials in z_n over the function field $\mathbb{F}(z_1, z_2, \cdots, z_{n-1})$. Consequently, there exists $a, b \in \mathbb{F}(z_1, z_2, \cdots, z_{n-1})[z_n]$ such that:

$$a\phi(f) + b\phi(g) = 1 \text{ in } \mathbb{F}(z_1, z_2, \cdots, z_{n-1})[z_n].$$

That is,

$$af + bg = 1 \text{ in } (R/\mathcal{M})(z_1, \cdots, z_{n-1})[z_n].$$

By the well known Hensel Lifting lemma we get that there exist $a^*, b^* \in R(z_1, \ldots, z_{n-1})[z_n]$ such that:

$$\begin{gathered} a^* f + b^* g = 1 \text{ in } (R/\mathcal{M}^t)(z_1, z_2, \cdots, z_{n-1})[z_n] \\ \text{which is } R(z_1, z_2, \cdots, z_{n-1})[z_n] \end{gathered}$$

Now by the assumption of the lemma:

$$\begin{aligned} & p \equiv 0 && \pmod f \\ \Rightarrow\quad & p = fq && \text{for some } q \text{ in } R[z_1, z_2, \cdots, z_{n-1}][z_n] \\ \mathit{also},\quad & p \equiv 0 && \pmod g \\ \Rightarrow\quad & fq \equiv 0 && \pmod g \\ \Rightarrow\quad & a^* fq \equiv 0 && \pmod g \ \text{ in } \ R(z_1, z_2, \cdots, z_{n-1})[z_n] \\ \Rightarrow\quad & q \equiv 0 && \pmod g \ \text{ in } \ R(z_1, z_2, \cdots, z_{n-1})[z_n] \\ \therefore\quad & p = fgh && \text{for some } h \text{ in } R(z_1, z_2, \cdots, z_{n-1})[z_n] \end{aligned}$$

Since, the leading coefficient of z_n in fg is in R^* and p, fg are in $R[z_1, z_2, \cdots, z_{n-1}][z_n]$, therefore by Gauss Lemma (see Lemma 4.3.1) we get that in fact $h \in R[z_1, z_2, \cdots, z_{n-1}][z_n]$ and so

$$p \equiv 0 \pmod{fg} \ \text{ in } R[z_1, z_2, \cdots, z_n].$$

□

4.4 Description of the Identity Test

4.4.1 Overview of the Algorithm

We now give an overview of our algorithm. The input is a $\Sigma\Pi\Sigma$ circuit $\mathcal{C}(x_1, \ldots, x_n)$ having an addition gate at the top with fanin k and computing a polynomial of total degree atmost d over a field $\mathbb{F}$. Our algorithm is recursive such that in each recursive call k reduces while the base ring (initially, it was $\mathbb{F}$) becomes larger. The intermediate larger rings that appear are all ensured to be local. The dimension of the base ring (over $\mathbb{F}$) increases by a factor of atmost d in each recursive call and thus, the complexity comes out to be $\mathit{poly}(d^k \cdot n)$ (assuming the field operations in $\mathbb{F}$ take constant time).

We will now demonstrate a snapshot of the algorithm. Let R be a local ring over the field $\mathbb{F}$ having maximal ideal $\mathcal{M}$. The circuit $\mathcal{C}(z_1, \ldots, z_n)$ in $R[z_1, \ldots, z_n]$ looks like:

$$\mathcal{C} = \beta_1 \cdot T_1 + \beta_2 \cdot T_2 + \ldots + \beta_k \cdot T_k$$

where each T_i is a product of linear forms

$$T_i = L_{i1} L_{i2} \cdots L_{id}$$

and where each L_{ij} is a linear form:

$$L_{ij} = a_{ij0} + a_{ij1}z_1 + a_{ij2}z_2 + \cdots + a_{ijn}z_n$$

for some $a_{ij1}, a_{ij2}, \cdots, a_{ijn} \in \mathbb{F}$ and $a_{ij0} \in \mathcal{M}$. We want to check if $\mathcal{C}$ computes the identically zero polynomial over R. Note that in each T_i, the coefficient of its leading monomial $\mathrm{Coeff}(LM(T_i), T_i)$ is in $\mathbb{F} \subseteq R^*$. We renumber the terms and ensure that

$$LM(T_1) \succeq LM(T_2) \succeq \ldots \succeq LM(T_k).$$

Suppose that T_1 factors over R into a product of *'coprime'* polynomials $p_1, \ldots, p_l$. We recursively verify that:

$$\mathcal{C} \equiv 0 \pmod{p_i} \quad \text{for } 1 \leq i \leq l$$

By our version of Chinese Remaindering Theorem for local rings we deduce that:

$$\mathcal{C} \equiv 0 \pmod{\prod_{i=1}^{l} p_i}$$

Our choice of the polynomials p_i ensures that the total degree of $\prod_{i=1}^{l} p_i(z_1, \ldots, z_n)$ is at least as large as that of $\mathcal{C}(z_1, \ldots, z_n)$. Finally by verifying that $\mathrm{Coeff}(LM(T_1), \mathcal{C})$, the coefficient of the leading monomial of T_1 in $\mathcal{C}(z_1, \ldots, z_n)$, is zero we deduce that $\mathcal{C}$ computes the identically zero polynomial over R.

Our choice of the polynomials p_i ensures two things:

i) There is an invertible linear transformation τ on the variables $\overline{z}$ such that it 'simplifies' the polynomial p_i:

$$\tau \circ p_i(z_1, \ldots, z_n) = (z_1 + m_1) \cdot (z_1 + m_2) \cdots (z_1 + m_s)$$

where, $m_j \in \mathcal{M}$. Thus, the ring $S_i := R[z_1]/(\tau \circ p_i)$ is a local ring.

ii) p_i divides T_1 and so $T_1 \equiv 0 \pmod{p_i}$. Thus $\tau \circ \mathcal{C}$ can be viewed as a $\Sigma\Pi\Sigma$ circuit with top fanin atmost $(k-1)$, total degree d and $(n-1)$ variate over the (larger) ring S_i. We can check $\mathcal{C} = 0 \pmod{p_i}$ by checking $\tau \circ \mathcal{C} = 0$ over S_i recursively.

4.4.2 The Algorithm

Input: The three inputs to the algorithm are:

- A local ring R of dimension r over a field $\mathbb{F}$ with maximal ideal $\mathcal{M}$. (In the initial call, $R = \mathbb{F}$ and $\mathcal{M} = \langle 0 \rangle$).
- A set of k coefficients $\langle \beta_1, \ldots, \beta_k \rangle$, where $k \geq 1$ and $\forall i : \beta_i \in R$.
- A set of k terms $\langle T_1, \ldots, T_k \rangle$. Each T_i is a product of d_i linear forms in n variables over the ring R. That is, each T_i is of the form $T_i = \prod_{j=1}^{d_i} l_{ij}$ and each $l_{ij} \in LF_{R/\mathbb{F}}(x_1, x_2, \ldots, x_n)$.

Output: The input parameters compute the following polynomial over the ring R:

$$p(x_1, \ldots, x_n) \stackrel{\text{def}}{=} \beta_1 T_1 + \ldots + \beta_k T_k$$

The output of the algorithm, $\mathbf{ID}(R, \langle \beta_1, \ldots, \beta_k \rangle, \langle T_1, \ldots, T_k \rangle)$, is YES iff

$$p(x_1, \ldots, x_n) \ = \ 0 \ \ .$$

Algorithm: $\mathbf{ID}(R,\ \langle \beta_1, \ldots, \beta_k \rangle,\ \langle T_1, \ldots, T_k \rangle)$:

Step 1: (Rearranging the terms.) By rearranging the terms if needed ensure that

$$LM(T_1) \succeq LM(T_i) \ \ \forall 2 \leq i \leq k.$$

Step 2: (Base case of one multiplication gate) If $k = 1$ then we need to check whether $\beta_1 T_1 = 0$ as a member of $R[x_1, x_2, \cdots, x_n]$. Since $LC(T_1)$ is a unit in $\mathbb{F}$, this happens if and only if $\beta_1 = 0$.

Step 3: (Verifying that $p(x_1, \ldots, x_n) \equiv 0 \ \ (mod\, T_1)$) We shall verify that T_1 divides $p(x_1, \ldots, x_n)$ by using recursion to verify that all the distinct coprime factors of T_1 divide $p(x_1, \ldots, x_n)$. Since T_1 is the product of linear forms over R, it can be written as the product of coprime factors, each factor being of the form

$$S = (l + m_1)(l + m_2) \ldots (l + m_t)$$

where $l \in \mathbb{F}[x_1, \ldots, x_n]$ is a linear form in n variables over $\mathbb{F}$. Now to verify that S divides $p(x_1, \ldots, x_n)$ do the following:

Step 3.1 (Applying a linear transformation.) Define a linear tranformation σ acting on the variables $x_1, \ldots, x_n$ such that σ sends $l \mapsto x_1$ and transforms $x_2, \ldots, x_n$ such that it is an invertible linear map. Now S divides $p(x_1, \ldots, x_n)$ if and only if $\sigma(S)$ divides $\sigma(p(x_1, \ldots, x_n))$.

Step 3.2 (Recursively verify $\sigma(S)$ divides $\sigma(p)$). Define the ring R' as

$$R' \overset{\text{def}}{=} R[x_1]/(\sigma(S)) \quad .$$

Note that $\sigma(T_1) \equiv 0 \pmod{\sigma(S)}$. For all i between 2 and k compute γ_i and T'_i such that:

$$\sigma(T_i) = \gamma_i T'_i \pmod{\sigma(S)} \text{ where } \gamma_i \in R' \text{ and } T'_i \in LF_{R'/\mathbb{F}}(x_2, \ldots, x_n).$$

Recursively call $\mathbf{ID}(R',\ \langle \beta_2\gamma_2, \ldots, \beta_k\gamma_k\rangle\ ,\ \langle T'_2, \ldots, T'_k\rangle)$. If the recursive call returns **NO** then output **NO** and exit.

Step 4: (Comparing coefficient of the highest monomial.) Compute the coefficient of $LM(T_1)$ in $p(x_1, \ldots, x_n)$ and output YES iff its zero.

4.4.3 Proof of Correctness

The proof of correctness is now straightforward. We continue using the notation set in the last subsection. The claim here is summarized as:

Theorem 4.4.1. *Let R be a local ring of dimension r over a field $\mathbb{F}$. Then*

$$\boldsymbol{ID}(R,\ \langle \beta_1, \ldots, \beta_k\rangle\ ,\ \langle T_1, \ldots, T_k\rangle)$$

returns YES iff $\beta_1 T_1 + \cdots + \beta_k T_k = 0$ in $R[x_1, \ldots, x_n]$. Furthermore, the time taken is $poly(nrd^k)$ assuming all the field operations can be done in constant time, where d is the maximum degree of any term.

Proof. **Time complexity.** Note that in all the recursive calls that

$$\mathbf{ID}(R,\ \langle \beta_1, \ldots, \beta_k\rangle\ ,\ \langle T_1, \ldots, T_k\rangle)$$

makes to $\mathbf{ID}(\cdot, \cdot, \cdot)$ the dimension of the base ring R increases by a factor of at most d whereas the value of k, the number of terms, decreases by one. Moreover there are at most d such recursive calls. Therefore, if $h(k, r)$ denotes the time taken by

$$\mathbf{ID}(R,\ \langle \beta_1, \ldots, \beta_k\rangle\ ,\ \langle T_1, \ldots, T_k\rangle)$$

then we have the following recurrence:

$$h(k, r) \quad \leq \quad d \cdot h(k-1, dr) + poly(ndrk)$$

Thus, we get that $h(k,r) = poly(nrd^k)$.

Correctness. We prove the correctess of the output of

$$\mathbf{ID}(R,\ \langle \beta_1, \ldots, \beta_k \rangle,\ \langle T_1, \ldots, T_k \rangle)$$

by induction on k:

Claim 4.4.1.1. *$\textbf{\textit{ID}}(R,\ \langle \beta_1, \ldots, \beta_k \rangle,\ \langle T_1, \ldots, T_k \rangle)$ returns YES iff*

$$\beta_1 T_1 + \ldots + \beta_k T_k = 0$$

Proof of Claim 4.4.1.1. The base case of the induction is when $k = 1$, handled and explained by Step 2.

Now we assume that $k \geq 2$ and that the claim is true for values smaller than k. Let $T_1 = S_1 \cdot S_2 \cdot \ldots \cdot S_m$. In Step 3 we verify that S_i divides $p(x_1, \ldots, x_n)$ for all $i \in [m]$. Then by lemma 4.3.4, we deduce that $T_1 = \prod_{i \in [m]} S_i$ divides $p(x_1, \ldots, x_n)$. Thus we get that

$$p(x_1, \ldots, x_n) = T_1 \cdot q(x_1, \ldots, x_n) \quad \text{for some} \quad q \in R[x_1, \ldots, x_n].$$

Since $LC(T_1)$ is a unit of R we have $LM(p) = LM(T_1) \cdot LM(q)$ and in particualar that $LM(p) \succeq LM(T_1)$. On the other hand, since $p = \sum_{i \in [k]} \beta_i T_i$ and $LM(T_1) \succeq LM(T_i)\ \ \forall i \in [k]$ we have that $LM(T_1) \succeq LM(p)$.

We therefore deduce that $LM(p) = LM(T_1)$. Finally in Step 4 we compute coefficient of $LM(T_1)$ in p and by the above observations it is the same as $LC(p)$. Now $p = 0$ over R iff $LC(p) = 0$ as required.

□ □

4.5 Discussion

In this chapter we generalized the well-known Chinese Remaindering lemma for polynomials to work even for polynomials over local rings and applied it to develop a deterministic polynomial-time identity test for $\Sigma\Pi\Sigma$ circuits with bounded top fanin. Ours was a *white-box derandomization* in that we had to look at the internal structure of the given circuit in order to determine if it computes the zero polynomial. There may well exist a stronger *black-box derandomization* of the identity testing problem. We first describe the notion of black-box derandomization as applicable to our problem and then mention a conjecture due to Manindra Agrawal [Agr05] which if true would imply such a black-box derandomization of the identity testing problem.

Black-box derandomization of Identity Testing. Let $\mathtt{C}_{k,d,n}$ be the set of all $\Sigma\Pi\Sigma$ algebraic circuits of top fanin k over a field $\mathbb{F}$ computing a polynomial of degree at most d in n variables over $\mathbb{F}$. It is well-known (cf. [Agr05]) that there exist a set $\mathtt{P}$ of $t = \mathrm{poly}(ndk)$ points $\{P_1, P_2, \ldots, P_t\} \subseteq \mathbb{F}^n$ such that any circuit $\mathcal{C} \in \mathtt{C}_{k,d,n}$ computes the zero polynomial if and only if $\mathcal{C}$ evaluates to zero at each of the points $P_1, P_2, \ldots, P_t$. Following the terminology of [Agr05], we call such a set $\mathtt{P}$ a *pseudo-random generator against the class of circuits* $\mathtt{C}_{k,d,n}$. The challenge then is to give an explicit (deterministic polynomial-time) construction of such a set $\mathtt{P}$.

A conjecture. Below we reproduce a conjecture which essentially claims that a set of points with roots of unity as coordinates is a pseudo-random generator for the class $\mathtt{C}_{k,d,n}$ of $\Sigma\Pi\Sigma$ algebraic circuits. In particular the conjecture implies an efficient deterministic identity testing algorithm for all $\Sigma\Pi\Sigma$-circuits.

Conjecture. (Agrawal, [Agr05].) Let n, d, k be natural numbers. Let $r \geq (n \cdot d \cdot k)^4$ be a prime. Let $\omega_r \in \mathbb{F}$ be a primitive r-th root of unity. Then the set of points $\mathtt{P} = \{P_1, P_2, \ldots, P_r\}$ given by

$$P_k \stackrel{\mathrm{def}}{=} (\omega_r^{k^0}, \omega_r^{k^1}, \ldots, \omega_r^{k^{n-1}})$$

is a pseudo-random generator against the class $\mathtt{C}_{k,d,n}$ of $\Sigma\Pi\Sigma$ algebraic circuits.

Chapter 5

Factoring Multivariate Polynomials over Finite Fields

Summary:
We consider the deterministic complexity of the problem of polynomial factorization over finite fields - given a finite field $\mathbb{F}_q$ and a polynomial $h(x, y) \in \mathbb{F}_q[x, y]$ compute the unique factorization of $h(x, y)$ as a product of irreducible polynomials. This problem admits a randomized polynomial-time algorithm and no deterministic polynomial-time algorithm is known. In this chapter, we give a deterministic polynomial-time algorithm that *partially* factors the input polynomial $h(x, y)$. The algorithm can be generalized to partially factor multivariate polynomials in an arbitary number of variables.

We now describe precisely the output of our partial factoring algorithm. Associated with every $\mathbb{F}_q$-irreducible factor $f(x, y)$ of $h(x, y)$ are two objects - its total degree n and the smallest extension field $\mathbb{F}_{q^d}$ of $\mathbb{F}_q$ over which $f(x, y)$ splits into absolutely irreducible factors. Collecting all the $\mathbb{F}_q$-irreducible factors of $h(x, y)$ which have the same degree and the same splitting field, we get a unique factorization of $h(x, y)$ into a product of "uniform polynomials" - polynomials whose component $\mathbb{F}_q$-irreducible factors all have the same degree and the same splitting field. It is this unique representation of $h(x, y)$ as a product of uniform polynomials that is outputted by our algorithm.

5.1 Introduction

A fundamental theorem of algebra states that polynomials over any field $\mathbb{F}$ admit a unique factorization into a product of (a finite number of) $\mathbb{F}$-irreducible factors. Computing this factorization for polynomials over various fields is a very well-studied problem in algorithmic number theory. For densely represented polynomials (that is, polynomials of degree n in m variables that are specified by giving all the $\binom{n+m}{m}$-possible coefficients of monomials), the problem of factoring multivariate polynomials is known to reduce to the problem of factoring univariate polynomials [Kal82]. For univariate polynomials over $\mathbb{Q}$, the field of rational numbers, Lenstra, Lenstra and Lovasz [LLL82] gave a deterministic polynomial-time algorithm.

Over finite fields, the problem admits random polynomial time algorithms [Ber67, Ber70, CZ81] but no deterministic polynomial-time algorithm is known. In a very interesting development, Kaltofen devised an algorithm that given an algebraic circuit computing a moderate degree polynomial in a large number of variables, computes its factorization in random polynomial time. Kaltofen's algorithm has been widely used in theoretical computer science with applications in list decoding of codes [GS99, Gur01] and hardness-randomness tradeoffs for arithmetic circuits [KI04].

The deterministic complexity of factoring polynomials over finite fields has also made partial progress. Berlekamp gave a deterministic algorithm for computing the distinct-degree factorization of univariate polynomials. This was subsequently generalized by Gao, Kaltofen and Lauder [GKL04] for deterministic distinct degree factorization of multivariate polynomials over finite fields. Motivated by the solvability problem to be tackled in the next chapter, we continue this line of work and develop a deterministic algorithm for partially factoring multivariate polynomials over finite fields. Moreover our algorithm can be parallelized so that the parallel time complexity is polylogarithmic in the degree of the input polynomial to be factored.

In order to describe the output of our algorithm we need to introduce some terms.

Definition 5.1.1. A bivariate polynomial $h(x, y) \in \mathbb{F}_q[x, y]$ is said to be *absolutely irreducible* if it is irreducible over $\mathbb{F}_q$ and remains irreducible over the algebraic closure $\overline{\mathbb{F}}_q$ of $\mathbb{F}_q$.

Example: For example, $(y^2 - x^3) \in \mathbb{F}_7[x, y]$ is absolutely irreducible whereas $(y^2 + x^2) \in \mathbb{F}_7[x, y]$ is irreducible over $\mathbb{F}_7$ but factors into $(y + \sqrt{-1}x)(y - \sqrt{-1}x)$ over the extension

$\mathbb{F}_{7^2} = \mathbb{F}_7(\sqrt{-1})$ and hence is not absolutely irreducible over $\mathbb{F}_7$.

Remark. Note that a univariate polynomial $f(x) \in \mathbb{F}_q[x]$ is absolutely irreducible if and only if it is a linear polynomial. To see this, observe that if $f(x) \in \mathbb{F}_q[x]$ is a univariate irreducible polynomial of degree $d \geq 2$ then then it splits properly over $\mathbb{F}_{q^d}$, and therefore cannot be absolutely irreducible.

The polynomial $h(x, y)$ has a unique factorization over the algebraic closure $\overline{\mathbb{F}}_q$ of $\mathbb{F}_q$. Now collect all the elements of $\overline{\mathbb{F}}_q$ that occur as the coefficient of some monomial $x^i y^j$ in some absolutely irreducible factor $g(x, y)$ of $h(x, y)$ over $\overline{\mathbb{F}}$. Since this is a finite set, all these coefficients lie in some finite extension $\mathbb{K}$ of $\mathbb{F}_q$. We will call the smallest such extension field $\mathbb{K}$ the splitting field of $h(x, y)$. We will denote by $dim_{\mathbb{F}_q}(h(x, y))$ the dimension of the splitting field $\mathbb{K}$ of $h(x, y)$ over $\mathbb{F}_q$. That is, $dim_{\mathbb{F}_q}(h(x, y)) \overset{\text{def}}{=} [\mathbb{K} : \mathbb{F}_q]$.

We will call a polynomial $f(x, y) \in \mathbb{F}_q[x, y]$ *a uniform polynomial* if any two of its $\mathbb{F}_q$-irreducible factors have the same total degree and the same splitting field $\mathbb{K}$. In this chapter, we build upon the distinct degree factorization algorithm of Gao, Kaltofen and Lauder [GKL04] to split a given polynomial $h(x, y) \in \mathbb{F}_q[x, y]$ into a product of uniform polynomials. We summarize our main result as a theorem:

Theorem 5.1.2. *[Uniform factoring] There exists a deterministic algorithm that on input a polynomial $h(x, y) \in \mathbb{F}_q[x, y]$ of degree n outputs*

$$\langle (h_1(x, y), n_1, d_1), \ldots, (h_k(x, y), n_k, d_k) \rangle$$

such that

$$h(x, y) = h_1(x, y) \cdot \ldots \cdot h_k(x, y)$$

where each $h_i(x, y)$ is a uniform polynomial consisting of $\mathbb{F}_q$-irreducible factors of degree n_i and splitting field $\mathbb{F}_{q^{d_i}}$.

The algorithm has a time complexity of poly$(n \cdot \log q)$. Moreover, the algorithm can be implemented parallelely to get a family of P -uniform circuits of depth poly$(\log n \cdot \log q)$ and size poly$(n \cdot \log q)$.

Note that the output of the algorithm of Theorem 5.1.2 is a refinement of the distinct degree factorization of $h(x, y)$ over $\mathbb{F}_q$.

We now give the overall idea behind our algorithm.

5.1.1 Basic Idea

The starting point of our algorithm is the procedure (due to Kaltofen [Kal82]) for reducing the problem of factoring bivariate polynomials to the problem of factoring univariate polynomials. Let $\mathbb{F}_q$ be a finite field and $h(x, y) \in \mathbb{F}_q[x, y]$ be a square-free bivariate polynomial of degree n that we wish to factor. By applying a suitable linear transformation if necessary, we can assume without loss of generality that $w(z) = h(z, 0)$ is square-free (cf. Kaltofen [Kal82]). Suppose we know an $\alpha \in \overline{\mathbb{F}}_q$ which is a root of some $\mathbb{F}_q$-irreducible factor $t(z)$ of $w(z)$. Let $R_t \stackrel{\text{def}}{=} \mathbb{F}_q[z]/\langle t(z)\rangle = \mathbb{F}_q(\alpha)$ be the splitting field of $t(z)$. Then by the squarefree-ness of $h(z, 0)$ there exists a unique (upto constant factors) minimal degree factor $h_{t(z)}(x, y) \in R_t[x, y]$ of $h(x, y)$ such that α is a root of $h_{t(z)}(z, 0)$. With this background in mind, Kaltofen's algorithm can be viewed as follows: using the root α having minimal polynomial $t(z)$ over $\mathbb{F}_q$, it simply writes down a system $\mathcal{R}_{t(z),m}$ of homogeneous linear equations over R_t whose solutions correspond to polynomials in $R_t[x, y]$ of degree at most m and which are multiples of $h_{t(z)}(x, y)$. Setting $m = n - 1$ and taking the gcd of all the polynomials corresponding to a basis of the solution space of $\mathcal{R}_{t(z),m}$ gives us the factor $h_{t(z)}(x, y) \in R_t[x, y]$ of $h(x, y)$.

Unfortunately the absence of a deterministic algorithm for univariate factoring over finite fields prevents us from obtaining irreducible factors of $w(z)$. Suppose that $v(z) \in \mathbb{F}_q[z]$ is *any* (not necessarily irreducible) factor of $w(z)$. As before, we construct the ring $R_v \stackrel{\text{def}}{=} \mathbb{F}_q[z]/\langle v(z)\rangle$ (note that R_v is no longer a field). We then view the element $\alpha' \in R_v$, $\alpha' \stackrel{\text{def}}{=} z \pmod{v(z)}$ as a *'pseudo-root'* of the polynomial $w(x) = h(x, 0) \in R_v[x]$. Proceeding as before, we write down a system $\mathcal{R}_{v(z),m}$ of homogeneous linear equations over R_v. We then ask the question - what do the solutions of $\mathcal{R}_{v(z),m}$ correspond to 'in reality'? Examining this question minutely, we deduce that by setting $v(z) = w(z)$ and varying m, the solutions of $\mathcal{R}_{w(z),m}$ can be used to factor out divisors of $h(x, y)$ having distinct degree or distinct splitting fields over $\mathbb{F}_q$.

Remark. Subsequently, Kaltofen [Kal85] essentially observed that $\mathcal{R}_{w(z),(n-1)}$ does **not** have a nontrivial solution if and only if $h(x, y)$ is absolutely irreducible. Combining this with efficient parallel algorithms for linear algebraic computations, he obtained a fast parallel deterministic algorithm for absolute irreducibility testing.

5.2 Mathematical machinery.

This section forms the core of this chapter. Its organized as follows - following tradition, we first introduce *nice* bivariate polynomials. We then examine how an $\mathbb{F}_q$-irreducible bivariate polynomial factors over various possible field extensions of $\mathbb{F}_q$. Next, we define some systems of linear equations $\mathcal{R}_{v(z),m}$, $\mathcal{F}_{v(z),m}$ and $\mathcal{B}_{v(z),m}$ and prove the basic properties of their solution spaces. Finally we show how these solution spaces can be used to obtain factors of $h(x,y)$. In this and the next section, we will use $h(x,y)$ for the reducible input polynomial to be factored and $f(x,y)$ for an $\mathbb{F}_q$-irreducible factor of $h(x,y)$.

5.2.1 Nice bivariate polynomials

Definition 5.2.1. A bivariate polynomial $f(x,y) \in \mathbb{F}_q[x,y]$ of total degree n is *nice* if $f(x,0)$ is squarefree and of degree n.

Note that the coefficient of x^i of a nice polynomial $f(x,y)$ as a polynomial in y has degree no more than $n-i$, in particular the leading coefficient of $f(x,y)$ with respect to x is in $\mathbb{F}_q$.

Also observe that a nice polynomial $f(x,y) \in \mathbb{F}_q[x,y]$ remains nice over any extension field $\mathbb{K}$ of $\mathbb{F}_q$ and that any factor of a nice polynomial is also a nice polynomial. By doing a square-free factorization of the input polynomial followed by a suitable linear transformation of the variables, the problem of general bivariate factoring can be reduced to factoring a nice bivariate polynomial (cf. Kaltofen [Kal82] for details).

Throughout the rest of this chapter we will use $\mathbb{F}_q$ to denote the input field and unless mentioned otherwise, all the algebras that we come across in this chapter will be over $\mathbb{F}_q$. Also we shall throughout use $h(x,y) \in \mathbb{F}_q[x,y]$ to denote the input polynomial to be factored.

5.2.2 How $\mathbb{F}_q$-irreducible bivariate polynomials behave over extensions of $\mathbb{F}_q$.

We will now examine how an $\mathbb{F}_q$-irreducible factor $f(x,y)$ of $h(x,y)$ factors over an extension field $\mathbb{F}_{q^d}$ of $\mathbb{F}_q$. We will show that over any extension field $f(x,y)$ splits into a product of *conjugate* factors and if the extension field happens to be isomorphic to $\mathbb{F}_q[z]/\langle v(z)\rangle$ where $v(z)$ is an irreducible factor of $f(z,0)$ then $f(x,y)$ splits into absolutely irreducible factors over it.

▮ *Conjugacy - an equivalence relation.*

Let $\mathbb{K}$ be a field extension of the finite field $\mathbb{F}_q$. Let $\phi \in Gal_{\mathbb{K}/\mathbb{F}_q}$ be an automorphism of $\mathbb{K}$. We extend ϕ to an automorphism of the ring $\mathbb{K}[x, y]$ in the natural way:

Definition 5.2.2. Let $\phi \in Gal_{\mathbb{K}/\mathbb{F}_q}$ be an automorphism of $\mathbb{K}$. Define the map $\phi : \mathbb{K}[x, y] \mapsto \mathbb{K}[x, y]$ as

$$\phi(f(x, y)) = \sum_{1 \leq k,l \leq n} \phi(a_{kl}) x^k y^l$$

where

$$f(x, y) = \sum_{1 \leq k,l \leq n} a_{kl} x^k y^l$$

Observe that the map $\phi : \mathbb{K}[x, y] \mapsto \mathbb{K}[x, y]$ is an automorphism of the ring $\mathbb{K}[x, y]$ that fixes the subring $\mathbb{F}_q[x, y]$. In particular,

- $\phi(f(x, y) + g(x, y)) = \phi(f(x, y)) + \phi(g(x, y))$
- $\phi(f(x, y) \cdot g(x, y)) = \phi(f(x, y)) \cdot \phi(g(x, y))$

We now define an equivalence relation on $\mathbb{K}[x, y]$ induced by such automorphisms of $\mathbb{K}[x, y]$.

Definition 5.2.3. Let $f(x, y), g(x, y) \in \mathbb{K}[x, y]$ be two bivariate polynomials. $g(x, y)$ is said to be a conjugate of $f(x, y)$ over $\mathbb{F}_q$, or an $\mathbb{F}_q$-conjugate of $f(x, y)$, if there exists an automorphism $\phi \in Gal_{\mathbb{K}/\mathbb{F}_q}$ such that $g(x, y) = \phi(f(x, y))$.

Observe that conjugacy is an equivalence relation on $\mathbb{K}[x, y]$.

▮ *Factorization of $\mathbb{F}_q$-irreducible polynomials over extension fields.*

Now consider a nice $\mathbb{F}_q$-irreducible polynomial $f(x, y) \in \mathbb{F}_q[x, y]$. Let $\mathbb{K} \supseteq \mathbb{F}_q$ be a finite field extension of $\mathbb{F}_q$. How does $f(x, y)$ factor over $\mathbb{K}$? We claim that all the $\mathbb{K}$-irreducible factors of $f(x, y)$ in $\mathbb{K}$ are in fact $\mathbb{F}_q$-conjugates of each other. In particular, all the $\mathbb{K}$-irreducible factors of $f(x, y)$ in $\mathbb{K}[x, y]$ are of equal degree.

Lemma 5.2.4. *Let $f(x, y) \in \mathbb{F}_q[x, y]$ be a nice $\mathbb{F}_q$-irreducible polynomial of total degree n. Let $\mathbb{K}$ be any finite field extension of $\mathbb{F}_q$. If $f_1(x, y) \in \mathbb{K}[x, y]$ and $f_2(x, y) \in \mathbb{K}[x, y]$ are any two $\mathbb{K}$-irreducible factors of $f(x, y)$ then $f_1(x, y)$ and $f_2(x, y)$ are $\mathbb{F}_q$-conjugates.*

Proof. For a polynomial $g(x,y) \in \mathbb{K}[x,y]$, define $H_g \leq Gal_{\mathbb{K}/\mathbb{F}_q}$ to be the subgroup of $Gal_{\mathbb{K}/\mathbb{F}_q}$ consisting of automorphisms in $Gal_{\mathbb{K}/\mathbb{F}_q}$ that fix $g(x,y)$. Since the galois groups of finite extensions of finite fields are cyclic groups, H_g must be a normal subgroup of $Gal_{\mathbb{K}/\mathbb{F}_q}$.

Let $g(x,y) \in \mathbb{K}[x,y]$ be a $\mathbb{K}$-irreducible factor of $f(x,y)$. Let the set of distinct cosets of H_g in $Gal_{\mathbb{K}/\mathbb{F}_q}$ be

$$Gal_{\mathbb{K}/\mathbb{F}_q}/H_g = \{H_g\phi_1, H_g\phi_2, \cdots H_g\phi_t\}$$

Then $\phi_1(g(x,y)), \phi_2(g(x,y)), \cdots \phi_t(g(x,y))$ are all the distinct conjugates of $g(x,y)$. We claim that the unique factorization of $f(x,y)$ into $\mathbb{K}$-irreducible polynomials over $\mathbb{K}$ is simply the product of all these distinct conjugates of $g(x,y)$. That is,

$$f(x,y) = \prod_{H_g\phi \in Gal_{\mathbb{K}/\mathbb{F}_q}/H_g} \phi(g(x,y)) \tag{5.1}$$

We first observe that any $\mathbb{F}_q$-conjugate of $g(x,y)$ is also a $\mathbb{K}$-irreducible factor of $f(x,y)$.

Claim 5.2.4.1. *Every conjugate of $g(x,y)$ is a $\mathbb{K}$-irreducible factor of $f(x,y)$.*

Proof. Since $g(x,y)|f(x,y)$, therefore $\exists g'(x,y) \in \mathbb{K}[x,y]$ such that $f(x,y) = g(x,y) \cdot g'(x,y)$. Suppose that ϕ is any automorphism in $Gal_{\mathbb{K}/\mathbb{F}_q}$. Applying ϕ to both sides we get:

$$\begin{aligned} &\phi(f(x,y)) = \phi(g(x,y)) \cdot \phi(g'(x,y)) \\ \Rightarrow &f(x,y) = \phi(g(x,y)) \cdot \phi(g'(x,y)) \\ \Rightarrow &\phi(g(x,y))|f(x,y) \end{aligned}$$

By the same reasoning $\phi(g(x,y)) \in \mathbb{K}[x,y]$ is $\mathbb{K}$-irreducible for if any $\widehat{g}(x,y) \in \mathbb{K}[x,y]$ is a proper divisor of $\phi(g(x,y))$ then $\phi^{-1}(\widehat{g}(x,y))$ is a a proper divisor $g(x,y)$, contradicting the $\mathbb{K}$-irreducibility of $g(x,y)$. Thus any conjugate of $g(x,y)$ is also an $\mathbb{K}$-irreducible factor of $f(x,y)$. □

Now $g(x,y)$ being $\mathbb{K}$-irreducible, is coprime to all $\mathbb{F}_q$-conjugates distinct from itself. Thus the rhs of equation (5.1) divides $f(x,y)$. Moreover the rhs of equation (5.1) is fixed by all the automorphisms in $Gal_{\mathbb{K}/\mathbb{F}_q}$. Since finite extensions of finite fields are normal extensions, so any polynomial in $\mathbb{K}[x,y]$ that is fixed by all the automorphisms in $Gal_{\mathbb{K}/\mathbb{F}_q}$ is in fact a polynomial in $\mathbb{F}_q[x,y]$. Hence the rhs of equation (5.1) is in fact a polynomial in $\mathbb{F}_q[x,y]$ that divides $f(x,y)$. By the $\mathbb{F}_q$-irreducibility of $f(x,y)$, we deduce that equation

(5.1) is indeed the unique factorization of $f(x,y)$. Thus all the $\mathbb{K}$-irreducible factors of $f(x,y)$ over $\mathbb{K}$ are precisely all the distinct conjugates of $g(x,y)$. □

Now consider an $\mathbb{F}_q$-irreducible polynomial $f(x,y) \in \mathbb{F}_q[x,y]$ that factors in the algebraic closure $\overline{\mathbb{F}}_q$ of $\mathbb{F}_q$. What is the splitting field of $f(x,y)$? Can we put a bound on the dimension of the splitting field over $\mathbb{F}_q$? Assuming that $f(x,y)$ is a nice polynomial, the following proposition shows that if $t(z)$ is an $\mathbb{F}_q$-irreducible factor of $f(z,0)$, then the splitting field of $f(x,y)$ is a subfield of the finite field $\mathbb{F}_q[z]/\langle t(z)\rangle$. In particular, if $f(z,0)$ has a root $\alpha \in \mathbb{F}_q$, then $f(x,y)$ must be absolutely irreducible.

Proposition 5.2.5. *Let $f(x,y) \in \mathbb{F}_q[x,y]$ be a nice $\mathbb{F}_q$-irreducible polynomial of total degree n whose splitting field is $\mathbb{F}_{q^d}$. Also let $t(z) \in \mathbb{F}_q[z]$ be an $\mathbb{F}_q$-irreducible factor of $f(z,0)$. Then $d|deg(t(z))$ and $f(x,y)$ breaks into absolutely irreducible factors over $\mathbb{K} := \mathbb{F}_q[z]/\langle t(z)\rangle$, each absolutely irreducible factor being of degree $m = \frac{n}{d}$.*

Proof. Let $g(x,y) \in \mathbb{K}[x,y]$ be a $\mathbb{K}$-irreducible factor of $f(x,y)$ in $\mathbb{K}[x,y]$. Suppose if possible that $g(x,y)$ is not absolutely irreducible but breaks further over some finite extension $\mathbb{L} \supset \mathbb{K}$.

Let H_g be as in lemma 5.2.4. By lemma 5.2.4

$$f(x,y) = \prod_{H_g\phi \in G/H_g} \phi(g(x,y)) \tag{5.2}$$

Let $\alpha \in \mathbb{K}$ be a root of the polynomial $t(z)$. We start with the observation that some $\mathbb{F}_q$-conjugate of α must be a root of $g(z,0)$. Since α is a root of $f(z,0)$ we have $(z-\alpha)|(f(z,0) = \prod_{H_g\phi\in G/H_g} \phi(g(z,0)))$. Being irreducible, $(z-\alpha)$ must divide one of the factors on the rhs. That is, $\exists \phi \in Gal_{\mathbb{K}/\mathbb{F}_q}$ such that $(z-\alpha)|\phi(g(z,0))$. Applying ϕ^{-1} to both sides, we get $(z-\beta)|g(z,0)$, where $\beta = \phi^{-1}(\alpha)$. This $\beta = \phi^{-1}(\alpha) \in \mathbb{K}$ is the required $\mathbb{F}_q$-conjugate of α that is a $\mathbb{K}$-root of the polynomial $g(z,0)$.

By lemma 5.2.4 the $\mathbb{L}$-irreducible factors of $g(x,y)$ in $\mathbb{L}[x,y]$ are all $\mathbb{K}$-conjugates. Let $g_1(x,y) \in \mathbb{L}[x,y]$ be such an $\mathbb{L}$-irreducible factor of $g(x,y)$ with $(z-\beta)$ dividing $g_1(z,0)$. Let $\psi \in Gal_{\mathbb{L}/\mathbb{K}}$ be such that $\psi(g_1(x,y)) \in \mathbb{L}[x,y]$ is another $\mathbb{L}$-irreducible factor of $g(x,y)$ distinct from $g_1(x,y)$. Now since $(z-\beta)|g_1(z,0)$, applying ψ on both sides we get that $(z-\psi(\beta))|\psi(g_1(z,0))$. But $\psi(\beta) = \beta$ and therefore $(z-\beta)$ divides two distinct coprime factors $g_1(z,0)$ and $\psi(g_1(z,0))$ of $g(z,0)$. This implies that $(z-\beta)^2$ divides $g(z,0)$ which is a contradiction since $f(z,0)$ and hence $g(z,0)$ are squarefree.

Thus the $\mathbb{K}$-irreducible factors of $f(x,y)$ are in fact absolutely irreducible. Hence there exists a subfield $\mathbb{F} \subseteq \mathbb{K}$ which is the splitting field of $f(x,y)$. Therefore $d = [\mathbb{F} : \mathbb{F}_q]$ divides $deg(t(z)) = [\mathbb{K} : \mathbb{F}_q] = [\mathbb{K} : \mathbb{F}][\mathbb{F} : \mathbb{F}_q]$.

By the definition of the splitting field of $f(x,y)$, the coefficients occuring in $g(x,y)$ lie in the field $\mathbb{F}$ and do not all lie in any proper subfield of $\mathbb{F}$. Hence $\mathbb{F}$ is precisely the subfield of $\mathbb{K}$ which is fixed by every automorphism in H_g. So

$$d = [\mathbb{F} : \mathbb{F}_q] = ord(Gal_{\mathbb{K}/\mathbb{F}_q}/H_g).$$

Further $ord(Gal_{\mathbb{K}/\mathbb{F}_q}/H_g)$ is the number of distinct absolutely irreducible factors of $f(x,y)$. Since all the absolutely irreducible factors of $f(x,y)$ are of the same degree, say m, we have

$$\begin{aligned} m.ord(Gal_{\mathbb{K}/\mathbb{F}_q}/H_g) &= deg(f(x,y)) \\ \Rightarrow\ m.d &= n \\ \Rightarrow\ m &= \frac{n}{d} \end{aligned}$$

□

This proposition means that if $f(x,y) \in \mathbb{F}_q[x,y]$ is a nice $\mathbb{F}_q$-irreducible polynomial and $t_1(z), t_2(z) \in \mathbb{F}_q[z]$ are any two $\mathbb{F}_q$-irreducible factors of $f(z,0)$ then the degree of an irreducible factor of $f(x,y)$ over $\mathbb{K}_1 \overset{\text{def}}{=} \mathbb{F}_q[z]/\langle t_1(z)\rangle$ is the same as the degree of an irreducible factor of $f(x,y)$ over $\mathbb{K}_2 \overset{\text{def}}{=} \mathbb{F}_q[z]/\langle t_2(z)\rangle$. This observation will be the key to our uniform-factoring algorithm.

5.2.3 Defining the linear systems.

We will now define some linear systems over R_v whose solutions capture different factors of $h(x,y)$. To be able to specify how these factors relate to a "seed polynomial" $v(z)$ we need to make the following definition.

Definition 5.2.6. Let R be any ring and let $v(z) \in R[z]$ be a univariate polynomial $f(x,y) \in R[x,y]$ be a bivariate polynomial. We will say that $f(x,y)$ *sits above* $v(z)$ if $v(z)$ divides $f(z,0)$.

We also extend the usual notion of squarefreeness of polynomials over fields to polynomials over arbitary rings.

Definition 5.2.7. Let R be any ring and $v(z) \in R[z]$ be a univariate polynomial over R. Let $v'(z) \in R[z]$ be the formal derivative of $v(z)$. We say that $v(z)$ is *squarefree* if $v(z)$ is coprime (see (2.1.11) for definition of coprimality) to $v'(z)$.

▮ *Fixing Some notation.*

We recall some of the quantities from the previous section and define and fix some other quantities that will be be used through the rest of this chapter.

As before, $h(x, y) \in \mathbb{F}_q[x, y]$ is a nice bivariate polynomial of degree n that we wish to factor. $w(z) \overset{\text{def}}{=} h(z, 0) \in \mathbb{F}_q[z]$ and $v(z) \in \mathbb{F}_q[z]$ is any factor of $w(z)$. Let the $\mathbb{F}_q$-irreducible factors of $v(z)$ be $v_j(z),\ 1 \leq j \leq r$.

$$R_v \overset{\text{def}}{=} \mathbb{F}_q[z]/\langle v(z)\rangle \cong \bigotimes_{j=1}^{r} \mathbb{F}_q[z]/\langle v_j(z)\rangle.$$

We will denote by π_{v_j} the projection of R_v onto the j-th component field,

$$R_{v_j} \overset{\text{def}}{=} \mathbb{F}_q[z]/\langle v_j(z)\rangle.$$

That is, for any $u \in R_v$,

$$\pi_{v_j}(u) \overset{\text{def}}{=} u \pmod{v_j}.$$

Note that every π_{v_j} extends in a natural manner to a homomorphism from polynomial rings over R_v to corresponding polynomial rings over R_{v_j}. We shall denote by B_v the Berlekamp subalgebra of R_v, defined as the subalgebra of R_v fixed by the automorphism $\phi : \zeta \mapsto \zeta^q$ of R_v.

The element $\alpha \in R_v$ is defined as $\alpha \overset{\text{def}}{=} z \pmod{v(z)}$ and it is an R_v-root of $h(x, 0) \in R_v[x]$.

We will now define three linear systems $\mathcal{R}_{v(z),m}$, $\mathcal{B}_{v(z),m}$ and $\mathcal{F}_{v(z),m}$. The solutions of each of these linear systems correspond to factors of $h(x, y) \in R_v[x, y]$ of degree at most m which sit above the polynomial $(x - \alpha)$. The difference is in which subring of $R_v[x, y]$ are these factors allowed to lie in, that is which subring of R_v do the coefficients come from.

The solutions of $\mathcal{F}_{v(z),m}$ are intended to capture factors (of degree at most m) which lie in the subring $\mathbb{F}_q[x, y]$ of $R_v[x, y]$. The solutions of $\mathcal{B}_{v(z),m}$ are intended to capture factors which lie in the subring $B_v[x, y]$ of $R_v[x, y]$. Finally, the solutions of $\mathcal{R}_{v(z),m}$ are intended to capture factors which lie in $R_v[x, y]$ itself.

Notational convention: In the rest of this chapter we will use $\mathtt{r}(x,y)$ to denote polynomials in $R_v[x,y]$, $\mathtt{b}(x,y)$ to denote polynomials in $B_v(x,y)$ and $\mathtt{f}(x,y)$ to denote polynomials in $\mathbb{F}_q[x,y]$.

Moreover for $m = \deg(h(x,y))$, the solutions of $\mathcal{R}_{v(z),m}$ are all going to be multiples of some particular well-defined polynomial $\mathtt{r}_{v(z)}(x,y) \in R_v[x,y]$. Similar thing is true for the linear systems $\mathcal{F}_{v(z),m}$ and $\mathcal{B}_{v(z),m}$. We will shortly define this factor $\mathtt{r}_{v(z)}(x,y) \in R_v[x,y]$ and its analogues. We prove a lemma first.

Lemma 5.2.8. *In the component field R_{v_j} of R_v, there exists a unique (upto constant multiples from R_{v_j}) minimal degree factor $\mathtt{r}_j(x,y) \in R_{v_j}[x,y]$ of $h(x,y)$ in $R_{v_j}[x,y]$ which sits above $(x - \pi_{v_j}(\alpha))$.*

Proof. **Existence.** Clearly $h(x,y) \in R_{v_j}[x,y]$ is itself a factor of $h(x,y)$ which sits above $\pi_{v_j}(x-\alpha)$ and therefore there does exist a minimal degree factor $\mathtt{r}_j(x,y)$ of $h(x,y)$ in $R_{v_j}[x,y]$ sitting above $\pi_{v_j}((x-\alpha))$.

Uniqueness. Note that $\mathtt{r}_j(x,y)$ must be an R_{v_j}-irreducible polynomial for if

$$\mathtt{r}_j(x,y) = \mathtt{r}_1(x,y) \cdot \mathtt{r}_2(x,y)$$

then either $\mathtt{r}_1(x,y)$ or $\mathtt{r}_2(x,y)$ sits above $(x - \pi_{v_j}(\alpha))$ and they are both factors of $h(x,y)$ in $R_{v_j}[x,y]$ of degree smaller than $\mathtt{r}_j(x,y)$, contradicting the assumption of the minimality of the degree of $\mathtt{r}_j(x,y)$. Suppose $\mathtt{r}'_j(x,y)$ is another factor of $h(x,y)$ sitting above $(x - \pi_{v_j}(\alpha))$, having the same degree as $\mathtt{r}_j(x,y)$. Then arguing as above, $\mathtt{r}'_j(x,y)$ is also irreducible in $R_{v_j}[x,y]$. Then their product $\mathtt{r}_j(x,y) \cdot \mathtt{r}'_j(x,y) \in R_{v_j}[x,y]$ must be factor of $h(x,y)$. But then $(x - \pi_{v_j}(\alpha))^2$ divides $\mathtt{r}_j(x,0) \cdot \mathtt{r}'_j(x,0)$ contradicting the squarefreeness of $h(x,0)$.

□

In a similar manner, by the $\mathbb{F}_q$-irreducibility of $v_j(z)$, there exists a unique $\mathbb{F}_q$-irreducible factor $\mathtt{b}_j(x,y) \in \mathbb{F}_q[x,y]$ such that $v_j(z)$ divides $\mathtt{b}_j(z,0)$. We will denote by $\mathtt{r}_{v(z)}(x,y)$ the unique element of $R_v[x,y]$ such that

$$\pi_{v_j}(\mathtt{r}_{v(z)}(x,y)) = \mathtt{r}_j(x,y) \forall 1 \leq j \leq r.$$

Analogously, we will denote by $\mathtt{b}_{v(z)}(x,y) \in B_v[x,y]$ the unique element of $B_v[x,y]$ such that

$$\pi_{v_j}(\mathtt{b}_{v(z)}(x,y)) = \mathtt{b}_j(x,y) \forall 1 \leq j \leq r.$$

Finally we define $\mathtt{f}_{v(z)}(x, y) \in \mathbb{F}_q[x, y]$ to be the polynomial

$$\mathtt{f}_{v(z)}(x, y) \overset{\text{def}}{=} \prod_{1 \leq j \leq r} \mathtt{b}_{v_j(z)}(x, y)$$

▮ *The linear systems $\mathcal{R}_{v(z),m}$, $\mathcal{F}_{v(z),m}$ and $\mathcal{B}_{v(z),m}$.*

The polynomials $(x - \alpha)$ and $\frac{w(x)}{(x-\alpha)}$ in $R_v[x]$ are formally coprime (since they are coprime in each of the projected fields). That is α is an ordinary root of $w(x) = h(x, 0)$ in $R_v = R_v[y]/\langle y \rangle$. We fix $k \in \mathbb{Z}_{>0}$ to be $k \overset{\text{def}}{=} 2n(n-1)$. By the well-known Hensel lifting lemma 2.1.12, there exists a unique $\alpha(y) = \alpha + \alpha_1 y + \alpha_2 y^2 + \ldots + \alpha_k y^k \in R_v[y]/\langle y^{k+1} \rangle$ such that

$$h(\alpha(y), y) = 0 \text{ (mod } \ y^{k+1}).$$

Moreover, $\alpha(y)$ is easily computed by iteratively solving linear equations over R_v.

Definition 5.2.9. The linear system $\mathcal{R}_{v(z),m}$ over R_v is defined to be the system

$$\sum_{i=0}^{m} u_i(y)\alpha(y)^i = 0 \ \text{ (mod } y^{k+1}) \tag{5.3}$$

with unknowns

$$u_i(y) \in R_v[y], \ deg(u_i(y)) \leq (m - i).$$

The definitions of the linear systems $\mathcal{B}_{v(z),m}$ and $\mathcal{F}_{v(z),m}$ are very similar except that the unknown polynomials are restricted to lie in the respective subrings of R_v.

Definition 5.2.10. The linear system $\mathcal{B}_{v(z),m}$ over B_v is defined to be the system

$$\sum_{i=0}^{m} u_i(y)\alpha(y)^i = 0 \ \text{ (mod } y^{k+1}) \tag{5.4}$$

with unknowns

$$u_i(y) \in B_v[y], \ deg(u_i(y)) \leq (m - i).$$

Definition 5.2.11. The linear system $\mathcal{F}_{v(z),m}$ over $\mathbb{F}_q$ is defined to be the system

$$\sum_{i=0}^{m} u_i(y)\alpha(y)^i = 0 \ \text{ (mod } y^{k+1}) \tag{5.5}$$

with unknowns

$$u_i(y) \in \mathbb{F}_q[y], \ deg(u_i(y)) \leq (m - i).$$

By *a solution* $\boldsymbol{r}(x,y)$ *of* $\mathcal{R}_{v(z),m}$ *in* $R_v[x,y]$ we will a mean a solution vector

$$(u_0(y), u_1(y), \ldots, u_m(y))$$

of the linear system $\mathcal{R}_{v(z),m}$, with $\mathtt{r}(x,y) \in R_v[x,y]$ being

$$\mathtt{r}(x,y) = \sum_{i=0}^{m} u_i(y)x^i \in R_v[x,y].$$

In an analogous manner we will identify solutions of $\mathcal{B}_{v(z),m}$ and $\mathcal{F}_{v(z),m}$ with bivariate polynomials $\mathtt{b}(x,y) \in B_v[x,y]$ and $\mathtt{f}(x,y) \in \mathbb{F}_q[x,y]$ respectively.

▮ *Properties of the linear systems for irreducible factors of $v(z)$.*

We will use $\mathcal{R}_{v_j(z),m}$ to denote the projection linear system $\mathcal{R}_{v(z),m}$ onto the j-th component:

$$\mathcal{R}_{v_j(z),m} \stackrel{\text{def}}{=} \pi_{v_j}(\mathcal{R}_{v(z),m}).$$

The projected linear systems $\mathcal{F}_{v_j(z),m}$ and $\mathcal{B}_{v_j(z),m}$ are defined analogously. We are now all set to prove the fundamental property of the solution space of these linear systems.

Proposition 5.2.12. *For all* $1 \le j \le r$*:*

1. *The projected linear system* $\mathcal{R}_{v_j(z),m}$ *has a non-zero solution if and only if*

$$\textsc{Deg}(\boldsymbol{r}_{v_j(z)}(x,y)) \le m.$$

Moreover, the gcd of all the polynomials in $R_{v_j}[x,y]$ *corresponding to a basis of the solution space of* $\mathcal{R}_{v_j(z),m}$ *is precisely the polynomial* $\boldsymbol{r}_{v_j(z)}(x,y) \in R_{v_j}[x,y]$.

2. *The projected linear system* $\mathcal{B}_{v_j(z),m}$ *has a non-zero solution if and only if*

$$\textsc{Deg}(\boldsymbol{b}_{v_j(z)}(x,y)) \le m.$$

Moreover, the gcd of all the polynomials in $B_{v_j}[x,y]$ *corresponding to a basis of the solution space of* $\mathcal{B}_{v_j(z),m}$ *is precisely the polynomial* $\boldsymbol{b}_{v_j(z)}(x,y) \in B_{v_j}[x,y]$.

3. *The projected linear system* $\mathcal{F}_{v_j(z),m}$ *has a non-zero solution if and only if*

$$\textsc{Deg}(\boldsymbol{f}_{v_j(z)}(x,y)) \le m.$$

Moreover, the gcd of all the polynomials in $\mathbb{F}_q[x,y]$ *corresponding to a basis of the solution space of* $\mathcal{F}_{v_j(z),m}$ *is precisely the polynomial* $\boldsymbol{f}_{v_j(z)}(x,y) \in \mathbb{F}_q[x,y]$.

Proof. The proofs of parts (ii) and (iii) are analogous to that of part (i) and we omit them for the sake of brevity. To emphasize that R_{v_j} is a field we will let $\mathbb{K}$ stand for it in the rest of this proof.

Existence of solution. Let $\mathtt{r}_{v_j(z)}(x,y) = v_0(y) + v_1(y)x + \ldots v_d(y)x^d$ where $d = \textsc{Deg}(\mathtt{r}_{v_j(z)}(x,y))$. Moreover $\mathtt{r}_{v_j}(x,y)$, being a factor of a nice polynomial $h(x,y)$ is itself a nice polynomial and so $\textsc{Deg}(v_i(y)) \leq (d-i)$. Now if $d \leq m$ then

$$(v_0(y), v_1(y), \cdots, v_d(y), 0, \ldots, 0)$$

is clearly a non-zero solution of the linear system $\mathcal{R}_{v_j(z),m}$. Conversely suppose that the system $\mathcal{R}_{v_j(z),m}$ has a nontrivial solution $g(x,y)$ with

$$g(x,y) := \sum_{i=0}^{m} u_i(y)x^i \in \mathbb{K}[x,y]$$

We claim that $\mathtt{r}_{v_j(z)}(x,y)$ must divide $g(x,y)$ thereby implying that $m \geq d$. Let

$$\rho(y) := \textsc{Resultant}_x(\mathtt{r}_{v_j(z)}(x,y), g(x,y)) \in \mathbb{K}[y]$$

Then $deg(\rho(y)) \leq (2n-1)n = k$. Then there exist polynomials $a(x,y), b(x,y) \in \mathbb{K}[x,y]$ such that

$$\rho(y) = a(x,y)\mathtt{r}_{v_j(z)}(x,y) + b(x,y)g(x,y) \tag{5.6}$$

Substituting $x := \alpha(y)$ in equation (5.6), we have

$$\rho(y) = 0 \pmod{y^{k+1}}.$$

But $deg(\rho(y)) \leq k$ and hence we must have that $\rho(y)$ is identically zero. Thus $gcd_x(\mathtt{r}_{v_j(z)}(x,y), g(x,y))$ is nontrivial whence by the irreducibility of $\mathtt{r}_{v_j(z)}(x,y)$ we deduce that $g(x,y)$ is a multiple of $\mathtt{r}_{v_j(z)}(x,y)$ as claimed. Thus we have shown that $\mathcal{R}_{v_j(z),m}$ has a non-zero solution if and only if

$$\textsc{Deg}(\mathtt{r}_{v_j(z}(x,y)) \leq m$$

and moreover $\mathtt{r}_{v_j(z)}(x,y)$ divides the bivariate polynomial in $\mathbb{K}[x,y]$ corresponding to any solution of $\mathcal{R}_{v_j(z),m}$.

The gcd of the basis vectors. Every solution of $\mathcal{R}_{v_j(z),m}$ corresponds to a bivariate polynomial over $\mathbb{K}$ in the natural way and let $g(x,y)$ be the gcd of all the basis polynomials which are solutions of $\mathcal{R}_{v_j(z),m}$. We must have that

$FactorR_{v_j(z)}(x, y)$ divides $g(x, y)$ because it divides every polynomial in the basis of $\mathcal{R}_{v_j(z),m}$. In the converse direction, observe that by definition, any solution of $\mathcal{R}_{v_j(z),m}$ is a $\mathbb{K}$-linear combination of the basis polynomials and therefore $g(x, y)$ divides any polynomial in the solution space. Since $\mathtt{r}_{v_j(z)}(x, y)$ is a solution of $\mathcal{R}_{v_j(z),m}$, we must have that $g(x, y)$ divides $\mathtt{r}_{v_j(z)}(x, y)$. Thus $\mathtt{r}_{v_j(z)}(x, y) = g(x, y)$ as was to be shown.

□

▮ *Using $\mathcal{F}_{v(z),n}$ to compute a factor of $h(x, y)$.*

Recall that n is the degree of $h(x, y)$ and now we set $m = n$ and look at solutions of $\mathcal{F}_{v(z),n}$. Note that the linear system $\mathcal{F}_{v(z),n}$ lies over the field $\mathbb{F}_q \subset R_v$ which is common to all the components R_{v_j}. Since $\text{DEG}(\mathtt{f}_{v_j(z)}(x, y)) \leq n$, by Proposition 5.2.12 all the projected linear systems $\mathcal{F}_{v_j(z),n}$ have a solution. In fact, among all factors $f(x, y)$ of $h(x, y)$ sitting above $v(z)$, $\mathtt{f}_{v(z)}(x, y)$ is the unique one with the minimal possible degree. By the above proposition, we can compute it efficiently by taking the gcd of all the basis polynomials in the solution space of $\mathcal{F}_{v(z),n}$. We record this discussion as a corollary.

Corollary 5.2.13. *Given a factor $v(z)$ of $h(z, 0)$ we can compute in deterministic polynomial time the unique minimal degree factor $f(x, y)$ of $h(x, y)$ such that $v(z)$ divides $f(z, 0)$.*

The linear systems $\mathcal{L}$ such as $\mathcal{R}_{v(z),m}$ and $\mathcal{B}_{v(z),m}$ will have nontrivial solutions in a projected component field R_{v_j} depending on whether the projected linear system $\mathcal{R}_{v_j(z),m}$ has a nontrivial solution there or not. The next proposition shows that if $\mathcal{L}$ has a nontrivial solution for some but not all the $v_j(z)$'s, then we can use the solutions of $\mathcal{L}$ to factor $v(z)$.

5.2.4 Factoring $v(z)$ using linear systems over R_v.

Recall that $v(z)$ is the product of r irreducible polynomials $v_j(z)$s.

Proposition 5.2.14. *Let $S \subseteq \{1, 2, \cdots r\}$ with the following property: the dimension over $\mathbb{F}_q$ of the solution space of the projected system $\mathcal{L}_{v_j}$ is non-zero if and only if $j \in S$. Then we can compute in deterministic polynomial time the nontrivial factor $(\prod_{j \in S} v_j(z))$ of $v(z)$.*

Proof. (We reproduce the following proof from Gao-Kaltofen-Lauder [GKL04].) Certainly any solution $\mathtt{r}(x, y)$ of $\mathcal{R}_{v(z),m}$ will be sent under the map π_{v_j} to a solution of $\mathcal{R}_{v_j(z),m}$ with entries in R_{v_j}. Moreover this solution will be non-zero if and only if $v_j(z)$ does not divide all of the coefficients in $\mathtt{r}(x, y)$ thought of as polynomials in $\mathbb{F}_q[z]$. Conversely any solution of $\mathcal{R}_{v_j(z),m}$ with entries in R_{v_j} can be lifted using the Chinese Remainder Theorem to a solution for $\mathcal{R}_{v(z),m}$ with entries in R_v.

Now compute a basis over $\mathbb{F}_q$ for the space of solutions in R_v of the linear system $\mathcal{R}_{v(z),m}$. We claim that the greatest commong divisor $g(z)$ say of $v(z)$ and the polynomials that occur as entries in the basis vectors (viewed as polynomials in z) is exactly $\prod_{j \notin S} v_j$.

To see this, suppose $j \in S$. Then there exists some non-zero solution $\mathtt{r}_j(x, y)$ of the linear system $\mathcal{R}_{v_j(z),m}$ which can be lifted to a non-zero solution $\mathtt{r}(x, y)$ of the linear system $\mathcal{R}_{v(z),m}$ as previously described. This solution $\mathtt{r}(x, y)$ has the property that at least one of the entries is not divisible by $v_j(z)$. This solution $\mathtt{r}(x, y)$ of $\mathcal{R}_{v_j(z),m}$ must lie in the $\mathbb{F}_q$-span of the basis vectors of the solution space of $\mathcal{R}_{v(z),m}$. Now if $v_j(z)$ divided all the entries in the basis vectors we would have that $v_j(z)$ divides all the entries of of all vectors in the solution space of $\mathcal{R}_{v(z),m}$ - a contradiction. Hence $v_j(z)$ does not divide $g(z)$. Now suppose, if possible, that $j \notin S$ and also that $v_j(z)$ does not divide $g(z)$. Then $v_j(z)$ does not divide all the entries in the basis vectors of the solution space of $\mathcal{R}_{v_j(z),m}$. Thus there exists at least one basis element $\mathtt{r}(x, y)$ which projects down to a non-zero solution of $\mathcal{R}_{v_j(z),m}$ under π_{v_j} - a contradiction. Thus $g(z)$ is as claimed.

Now one may compute the factor $g(z)$ in deterministic polynomial time using only a deterministic algorithm for computing the solution space over $\mathbb{F}_q$ of the linear system $\mathcal{R}_{v(z),m}$ and the euclidean algorithm for greatest common divisors of univariate polynomials. Moreover, this can be done efficiently in parallel.

□

5.3 The Algorithm.

Proposition 5.3.1. *Let $m \geq 1$ be a natural number and $h(x, y) \in \mathbb{F}_q[x, y]$ a nice polynomial. There is a deterministic polynomial-time algorithm that given $\langle \mathbb{F}_q, h(x, y), m \rangle$ obtains the product of all the $\mathbb{F}_q$-irreducible factors of $h(x, y)$ having degree at most m.*

Proof. Let $f(x, y) \in \mathbb{F}_q[x, y]$ be the product of all $\mathbb{F}_q$-irreducible factors of $h(x, y)$ having degree at most m. Set $v(z)$ to be $h(z, 0)$. We claim that the projected linear system $\mathcal{B}_{v_j(z),m}$ has a solution in R_{v_j} if and only if $v_j(z)$ divides $f(z, 0)$.

($\Rightarrow$) By Proposition 5.2.12, $\mathtt{b}_{v_j(z)}(x,y) \in \mathbb{F}_q[x,y]$ which is an $\mathbb{F}_q$-irreducible factor of $h(x,y)$ is a solution of the projected system $\mathcal{B}_{v_j(z),m}$. Moreover from the definition of the linear system $\mathcal{B}_{v_j(z),m}$, $\mathtt{b}_{v_j(z)}(x,y)$ has degree at most m. Therefore $\mathtt{b}_{v_j(z)}(x,y)|f(x,y)$. But $v_j(z)|\mathtt{b}_{v_j(z)}(z,0)$ and therefore $v_j(z)|f(z,0)$ as required.

($\Leftarrow$) Since $v_j(z)|f(z,0)$, by the squarefree-ness of $f(z,0)$, there exists a unique $\mathbb{F}_q$-irreducible factor $g(x,y)$ of $f(x,y)$ of degree at most m such that $v_j(z)|g(z,0)$. From the definition of the linear system $\mathcal{B}_{v_j(z),m}$ this polynomial $g(x,y)$ is clearly a solution of $\mathcal{B}_{v_j(z),m}$.

By Proposition 5.2.14, we can recover $f(z,0)$ and using this seed factor of $h(z,0)$ as input to the algorithm of Proposition 5.2.13, we can compute $f(x,y)$ in deterministic polynomial time.

□

Given any polynomial $h(x,y)$ of degree we obtain by the above algorithm a factor $f(x,y)$ consisting of $\mathbb{F}_q$-irreducible factors of degree at most $m := \frac{n}{2}$. Recursively repeating this process (in parallel) on the polynomials $f(x,y)$ and $\frac{h(x,y)}{f(x,y)}$, we obtain a distinct degree factorization of $h(x,y)$ in deterministic time poly$(n \cdot \log q)$. Moreover implementing all the fundamental linear-algebraic operations over $\mathbb{F}_q$ in parallel we can do this in parallel time poly$(\log n \cdot \log q)$.

Proposition 5.3.2. *Let $m, d \geq 1$ be natural numbers and $h(x,y) \in \mathbb{F}_q[x,y]$ a nice polynomial, each of whose $\mathbb{F}_q$-irreducible factors has degree at most m. There is a deterministic polynomial-time algorithm given $\langle \mathbb{F}_q, h(x,y), m, d\rangle$ obtains the product of all the $\mathbb{F}_q$-irreducible factors of $h(x,y)$ having a splitting field of size at least q^d.*

Proof. Let $f(x,y) \in \mathbb{F}_q[x,y]$ be the product of all $\mathbb{F}_q$-irreducible factors of $h(x,y)$ having a splitting field of size at least q^d. Set $v(z)$ to be $h(z,0)$ and $k = \frac{m}{d}$. We claim that the projected linear system $\mathcal{R}_{v_j(z),k}$ has a solution in R_{v_j} if and only if $v_j(z)$ divides $f(z,0)$.

($\Rightarrow$) By Proposition 5.2.12, $\mathtt{r}_{v_j(z)}(x,y) \in R_{v_j}[x,y]$ which is an absolutely irreducible factor of $h(x,y)$ is a solution of the projected system $\mathcal{R}_{v_j(z),k}$. Moreover from the definition of the linear system $\mathcal{R}_{v_j(z),k}$, $\mathtt{r}_{v_j(z)}(x,y)$ has degree at most k. Also $\mathtt{b}_{v_j(z)}(x,y) \in \mathbb{F}_q[x,y]$ is an $\mathbb{F}_q$-irreducible factor of $h(x,y)$ and since all $\mathbb{F}_q$-irreducible factors of $h(x,y)$ have degree m, therefore $\mathtt{b}_{v_j(z)}(x,y)$ also has degree m. Now, $\mathtt{r}_{v_j}(x,y)$ divides $\mathtt{b}_{v_j(z)}(x,y) \in \mathbb{F}_q[x,y]$, an $\mathbb{F}_q$-irreducible factor of $h(x,y)$. By Proposition 5.2.5,

$$\begin{aligned}\text{dimension of } \mathtt{b}_{\mathrm{v_j(z)}}(\mathrm{x,y}) &= \textsc{Deg}(\mathtt{b}_{v_j(z)}(x,y))/\textsc{Deg}(h_{v_j(z)}(x,y))\\ &= m/\textsc{Deg}(h_{v_j(z)}(x,y))\\ &\geq d\end{aligned}$$

Therefore $\mathtt{b}_{v_j(z)}(x,y)|f(x,y)$. But $v_j(z)|\mathtt{b}_{v_j(z)}(z,0)$ and therefore $v_j(z)|f(z,0)$ as required.

($\Leftarrow$) Since $v_j(z)|f(z,0)$, by the squarefree-ness of $f(z,0)$, there exists a unique $\mathbb{F}_q$-irreducible factor $g(x,y)$ of $f(x,y)$ of degree at most m such that $v_j(z)|g(z,0)$. From the definition of the linear system $\mathcal{B}_{v_j(z),m}$ this polynomial $g(x,y)$ is clearly a solution of $\mathcal{B}_{v_j(z),m}$.

By Proposition 5.2.14, we can recover $f(z,0)$ and using this seed factor of $h(z,0)$ as input to the algorithm of Proposition 5.2.13, we can compute $f(x,y)$ in deterministic polynomial time.

□

Given a $h(x,y)$ and m as in the statement of this proposition and setting $d = \frac{m}{2}$, we obtain by the above algorithm a factor $f(x,y)$ consisting of $\mathbb{F}_q$-irreducible factors of dimension at most $d := \frac{m}{2}$. Recursively repeating this process (in parallel) on the polynomials $f(x,y)$ and $\frac{h(x,y)}{f(x,y)}$, we obtain a uniform factorization of $h(x,y)$ in deterministic time $\mathrm{poly}(n \cdot \log q)$, and in parallel time $\mathrm{poly}(\log n \cdot \log q)$.

This completes the proof of Theorem 5.1.2.

Chapter 6

Solvability of Polynomial Equations over Finite Fields

Summary:
We investigate the complexity of the following polynomial solvability problem: given a finite field $\mathbb{F}_q$ and a set of polynomials $f_1, f_2, \cdots, f_m \in \mathbb{F}_q[x_1, x_2, \cdots, x_n]$ of total degree at most d determine the $\mathbb{F}_q$-solvability of the system $f_1 = f_2 = \cdots = f_m = 0$. This problem is easily seen to be NP-complete even when the field size q is as small as 2 and the degree of each polynomial is bounded by $d = 2$. Here we investigate the deterministic complexity of this problem when the number of variables n in the input is bounded. We show that there is a *deterministic* algorithm for this problem whose running time, for any fixed n, is bounded by a polynomial in d, m and $\log q$.

6.1 Introduction

6.1.1 Motivation

Studying the solution set of a system of polynomial equations is one of the main preoccupations of mathematics. Indeed, three of the most celebrated results of the twentieth century pertain to the solutions of polynomial equations:

- **Weil's Theorem**, also known as the Riemann Hypothesis for curves over finite fields, which gives bounds on the number of rational points on smooth projective curves over finite fields.

- **Falting's Theorem** which states that any curve over $\mathbb{Q}$, the field of rational numbers, of genus greater than 1 has only a finite number of rational points.

- **Wiles' Theorem** which states that the curve $x^n + y^n = 1$ has no nontrivial $(xy \neq 0)$ solution over the field of rational numbers for $n \geq 3$.

This motivates the study of the corresponding computational problems - given a set of polynomials over a field $\mathbb{F}$:

- **Solvability:** Determine whether there exists a common zero of the polynomials.

- **Counting solutions:** Determine the number of common zeroes.

- **Computing a solution:** Compute a common zero, if it exists.

One gets different computational problems depending on whether one is looking for common zeroes in $\mathbb{F}$ itself (i.e. $\mathbb{F}$-rational points) or in the algebraic closure $\overline{\mathbb{F}}$ of $\mathbb{F}$. The decidability of the solvability problem for rational points over $\mathbb{Q}$ is an intensively investigated open problem (Poonen [Poo02] gives a survey). In this chapter we consider the solvability problem for rational points over finite fields. We give a deterministic polynomial-time algorithm for the solvability problem over finite fields when the number n of variables in the system is bounded. Our results can be viewed as the natural algorithmic outcome of Weil's theorem. Indeed using Weil's bounds, we get an algorithm with similar complexity for the approximate counting version of the problem. We remark here that given a set of polynomial equations over $\mathbb{Q}$, the field of rational numbers, one can deduce certain properties of the solution set by looking at the reduction of the system of equations modulo p for various primes p and use this information to deduce global values of those properties of the solution set over $\mathbb{Q}$. For certain particularly special sets of polynomial equations over $\mathbb{Q}$, it might be sufficient to verify solvability modulo lots of primes p in order to deduce the existence of a solution over $\mathbb{Q}$. We make one such conjecture in the chapter on open problems. In general, however, there exist polynomials which have lots of $\mathbb{F}_p$-solutions for all primes p but no solution over the rational numbers. Nevertheless, given such a set of equations over $\mathbb{Q}$, one can determine almost all the *geometric properties* such as the numer of $\mathbb{C}$-irreducible components, their dimension and degree of the solution set by looking at the solution set modulo p (see Huang and Wong, [HW00] for details).

Our basic solvability algorithm can be extended in two ways to give more information about the algebraic set $\mathtt{X}$ defined by the given set of polynomials over $\mathbb{F}_q$. We also get efficient deterministic algorithms for:

- Approximating the number of $\mathbb{F}_q$-points on $\mathtt{X}$.
- Computing the number of irreducible components of $\mathtt{X}$ together with the degree and dimension of each such irreducible component.

6.1.2 Problem Definition

Here we are interested in the computational complexity of the solvability problem over the domain of finite fields. The most general version of the polynomial system problem is:

Problem - Existence of solution to a polynomial system (Solvability)

Input. The input is $\langle \mathbb{F}_q, f_1, f_2, \cdots f_m \rangle$ where : (i) $\mathbb{F}_q$ is a finite field with $q = p^r$ being a prime power. The finite field can be specified in the usual way by giving a prime p and an irreducible polynomial of degree r over $\mathbb{F}_p$. (ii) $f_1, f_2, \cdots, f_m \in \mathbb{F}_q[x_1, x_2, \cdots, x_n]$ are m polynomials in the n variables $x_1, x_2, \cdots, x_n$ with coefficients coming from the field $\mathbb{F}_q$. The polynomials are specified using the *dense* representation. That is, a polynomial of degree d in n variables over $\mathbb{F}_q$ has input size $\binom{d+n}{d} \cdot \log q$.

Question. Does there exist a point $(a_1, a_2, \cdots, a_n) \in \mathbb{F}_q^n$ such that

$$f_i(a_1, a_2, \cdots, a_n) = 0 \text{ for all } 1 \leq i \leq m$$

The general polynomial system problem is easily seen to be NP-complete even over a field as small as $\mathbb{F}_2$ and even when all the polynomials in the specified system are of total degree at most 2. This suggests that the problem becomes intractable when the number of variables is large. We examine the complexity of this problem when the number n of variables in the input system is bounded. Huang and Wong [HW96] give a randomized polynomial time (**Z**PP) algorithm for the bounded-variable version of this problem leaving the determintistic complexity unresolved. Our contribution to this problem is to give a *deterministic* polynomial-time algorithm. Moreover, our algorithm works for arbitary finite fields and not just prime fields.

Remark. Consider the slightly more general problem - given a finite field $\mathbb{F}_q$ and polynomials $f_1, f_2, \ldots, f_m$ and $g_1, g_2, \ldots, g_l \in \mathbb{F}_q[\bar{\mathbf{x}}]$ in n variables over $\mathbb{F}_q$, determine if there exists a point $\bar{\mathbf{a}} \in \mathbb{F}_q^n$ such that

$$f_1(\bar{\mathbf{a}}) = \ldots = f_m(\bar{\mathbf{a}}) = 0 \text{ and } g_1(\bar{\mathbf{a}}) \neq 0, \ g_2(\bar{\mathbf{a}}) \neq 0 \ \ldots, \ g_l(\bar{\mathbf{a}}) \neq 0$$

Such an apparently more general problem, involving both equations and 'inequations' over a field is easily seen to reduce to the solvability problem via what is known as the "'Rabinovich trick"' - introduce a new variable y and determine the $\mathbb{F}_q$-solvability of the following system of equations instead:

$$f_1(\bar{\mathbf{x}}) = f_2(\bar{\mathbf{x}}) = \ldots = f_m(\bar{\mathbf{x}}) = 0, \;\; y \cdot g_1(\bar{\mathbf{x}}) \cdot \ldots \cdot g_l(\bar{\mathbf{x}}) = 1$$

Remark. Let $f(x) = \frac{g(x)}{h(x)} \in \mathbb{F}_q(x)$ be a rational function over $\mathbb{F}_q$ with $\gcd(g(x), h(x)) = 1$. Then $f(x)$ induces a partial mapping $\mathbb{F}_q \mapsto \mathbb{F}_q$ via the map $a \mapsto f(a)$ for $a \in \mathbb{F}_q$. If $f(x)$ is total and bijective then $f(x)$ is called a *permutation function* over $\mathbb{F}_q$. In the special case that $h(x) = 1$, so that $f(x) = g(x) \in \mathbb{F}_q[x]$, it is called a *permutation polynomial* over $\mathbb{F}_q$. Permutation functions have been investigated theoretically [Wil68, Mac67, DL63, BD66, Hay67, Coh70], applied in cryptography [LM83] and the complexity of recognizing them dealth with [Shp92, Gat91, Gat89, MG95]. Shparlinski [Shp92] gave a deterministic **superpolynomial**-time algorithm for this problem while Ma and Gathen [Gat91, MG95] gave an efficient **randomized** algorithm. The existence of an efficient deterministic algorithm was open.

Now note that $f(x) = \frac{g(x)}{h(x)}$ is a permutation function if and only if $f(x)$ is total ($h(x) = 0$ has no $\mathbb{F}_q$-solution) and

$$g(x)h(y) - g(y)h(x) = 0, \quad x \neq y$$

has no $\mathbb{F}_q$-solution. Thus, by the remark above, recognizing permutation functions boils down to the solvability problem in 3 variables. Our deterministic solvability algorithm now implies an efficient deterministic algorithm for recognizing permutation functions and thus resolves the deterministic complexity of this problem as well.

6.1.3 Our results

We summarize our main result as a theorem:

Theorem 6.1.1. *There exists a deterministic algorithm which solves the decision version of the Solvability problem on an input consisting of a finite field $\mathbb{F}_q$ and polynomials $f_1, f_2, \cdots, f_m \in \mathbb{F}_q[x_1, x_2, \cdots, x_n]$ of total degree bounded by d in time poly($d^{c_n} \cdot m \cdot \log q$), where c_n is a constant that depends on n alone and is of size $n^{O(n)}$. Moreover, the algorithm can be implemented parallely to get a family of P -uniform circuits of depth poly($c_n \cdot \log d \cdot \log m \cdot \log q$) and size poly($d^{c_n} \cdot m \cdot \log q$) for the solvability problem.*

The basic algorithm for solvability can be easily extended to get an approximation algorithm of the same complexity for the counting version of the problem. More precisely, the algorithm calculates two non-negative integers N and D, such that $|\#V - Nq^D|$ is bounded by $d^{c_n} q^{D-1/2}$ for some constant c_n that depends on n alone, where $\#V$ denotes the number of common $\mathbb{F}_q$-solutions of the given set of polynomials.

6.1.4 The Idea

The input polynomials with coefficients from $\mathbb{F}_q$ describe an algebraic closed set $\mathtt{X}$. Our aim is to determine if the given closed set $\mathtt{X}$ over the given field $\mathbb{F}_q$ has any $\mathbb{F}_q$-rational point or not. The basic idea is to decompose the given closed set $\mathtt{X}$ into a union of (possibly reducible) closed sets $\mathtt{X}_i$, each $\mathtt{X}_i$ being birational to a hypersurface $\mathtt{Y}_i$. Now Weil's theorem and its generalizations [Sch74, CM03, CM04] imply the abundance of $\mathbb{F}_q$-rational points on any absolutely irreducible $\mathbb{F}_q$-hypersurface. We use the partial factoring algorithm developed in the previous chapter to determine, for each i, if any of the component $\mathbb{F}_q$-irreducible hypersurfaces of $\mathtt{Y}_i$ is absolutely irreducible or not. If $\mathtt{Y}_i$ happens to have an absolutely irreducible $\mathbb{F}_q$-factor, we use Weil's theorem to deduce an abundance of rational points on $\mathtt{Y}_i$ and, via the birational correspondence, on $\mathtt{X}_i$ as well. Otherwise a rational point on $\mathtt{X}_i$, if it exists, must lie on a closed proper subset of $\mathtt{X}_i$. We compute this subset of $\mathtt{X}_i$ and determine the existence of a rational point on it recursively.

Comparison with previous algorithms. Our approach parallels that of Huang and Wong ([HW99]) and it can be viewed as a deterministic modification of their algorithm. Indeed, [HW99] remark that their method actually gives a deterministic reduction to univariate factorization so that the only point that prevents their algorithm from being deterministic is the lack of a deterministic polynomial time algorithm for factoring univariate polynomials over finite fields. *The key contribution of our work on this problem is to observe that as far as the decision version of the problem of solvability is concerned, we do not need to completely factor the multivariate polynomials that arise during this computation process.* In both the works, the algorithm consists of two phases: we first decompose the algebraic closed set corresponding to the given set of equations and reduce the problem to the case of hypersurfaces and then determine the existence of a rational point on the hypersurface by testing for absolutely irreducibility. The difference is that in the first phase, while their algorithm decomposes the set into $\mathbb{F}_q$-irreducible components, the output components of the first phase in our case need not be $\mathbb{F}_q$-irreducible. Our contribution here is to observe that the operations involved and the proofs which hold for irreducible components and their corresponding fields go through with minor modifications when we are working with reducible algebraic sets and their corresponding rings. In the second phase, instead of testing the absolute irreducbility of an $\mathbb{F}_q$-irreducible polynomial, our algorithm uses the output of the partial factoring algorithm developed in the previous chapter. Moreover, they use efficiently parallelizable subroutines developed earlier by Grigoriev, Chistov, et al in order to ensure that the algorithm is efficiently parallelizable with respect to d and m. *We give a self-contained treatment here which preserves this parallelism while eliminating randomness.* Finally, the algorithm in [HW99] works only over prime fields while our algorithm works over all finite fields $\mathbb{F}_q$, even those with a small characteristic p. The difficulty in going from prime fields (q is prime) to general finite fields (q is prime power) is the existence of polynomials $f(x_1, \ldots, x_n) \in \mathbb{F}_q[x_1, \ldots, x_n]$ of degree $d \ll q$ which are squarefree and yet not separable. For example, $f(x_1, x_2) = x_2^p - x_1$ viewed as a univariate polynomial in x_2 over the function field $\mathbb{F} \overset{\text{def}}{=} \mathbb{F}_q(x_1)$ is squarefree and yet has repeated roots in the algebraic closure of $\mathbb{F}$. We overcome this difficulty by observing that a random linear transformation $\sigma \in \mathbb{F}_q^{n\times n}$ on the variables transforms a square-free non-separable polynomial $f(\bar{\mathbf{x}})$ to a separable polynomial in x_n. We then replace $f(\bar{\mathbf{x}})$ by $\sigma(f(\bar{\mathbf{x}}))$ in our computations and work with this transformed polynomial instead.

We flesh out this basic idea in more detail in a later section, after introducing the appropriate terminology and proving some basic facts.

6.2 Basic Algebraic Geometry with Examples

In this section we give a very quick overview of some basic facts from algebraic geometry and introduce the terminology to be used. For proofs see any basic text in algebraic geometry such as Shafarevich [Sha94]. We then give some representative examples.

Algebraic Closed Sets. Let $\mathbb{F}$ be a field. The algebraic closure of $\mathbb{F}$ will be denoted by $\overline{\mathbb{F}}$. A *closed algebraic set over* $\mathbb{F}$ is a subset $\mathtt{X}$ of $\overline{\mathbb{F}}^n$ consisting of all common zeroes of a finite number of polynomials in n variables with coefficients in $\mathbb{F}$. When the field $\mathbb{F}$ is understood from context we will simply refer to $\mathtt{X}$ as a closed algebraic set or just a closed set. A $\mathbb{F}$-*rational point of* $\mathtt{X}$ is a point $P \in \mathtt{X}$ all of whose coordinates are in $\mathbb{F}$.

We shall write $f(\bar{\mathbf{x}})$ to denote a polynomial in n variables, allowing $\bar{\mathbf{x}}$ to stand for the n-tuple of variables $(x_1, x_2, \ldots, x_n)$. If a closed set $\mathtt{X}$ consists of all common zeroes of polynomials $f_1(\bar{\mathbf{x}}), \ldots, f_m(\bar{\mathbf{x}})$, then we refer to $f_1(\bar{\mathbf{x}}) = \cdots = f_m(\bar{\mathbf{x}}) = 0$ as the equations of the set $\mathtt{X}$. We say that $\mathtt{X}$ is a *hypersurface* when it is specified by a single equation $(m = 1)$. Observe that a point $P = (a_1, \ldots, a_n) \in \overline{\mathbb{F}}^n$ belongs to the closed algebraic set $\mathtt{X}$ if and only if for all $i \in [m]$, $f_i(x_1 + a_1, \ldots, x_n + a_n)$ has no constant term. $P \in \mathtt{X}$ is said to be a *singular point* of $\mathtt{X}$ iff for all $i \in [m]$, $f_i(x_1 + a_1, \ldots, x_n + a_n)$ has no constant as well as no linear terms. We will say that a closed set $\mathtt{Y}$ is a *singular closed subset of* $\mathtt{X}$ iff every point $P \in \mathtt{Y}$ is a singular point.

A closed algebraic set $\mathtt{X}$ is said to be *reducible* if there exist proper closed subsets $\mathtt{X}_1, \mathtt{X}_2 \subsetneq \mathtt{X}$ such that $\mathtt{X} = \mathtt{X}_1 \cup \mathtt{X}_2$. Otherwise $\mathtt{X}$ is *irreducible*. An irreducible algebraic closed set $\mathtt{X}$ is also referred to as a *variety*.

It is a fundamental theorem in algebraic geometry that any closed agebraic set $\mathtt{X}$ is a finite union of irreducible algebraically closed sets. Now if $\mathtt{X} = \bigcup \mathtt{X}_i$ is an expression of $\mathtt{X}$ as a finite union of irreducible closed sets, and if $\mathtt{X}_i \subseteq \mathtt{X}_j$ then we can delete $\mathtt{X}_i$ from the representation. Repeating this several times, we arrive at a representation $\mathtt{X} = \bigcup \mathtt{X}_i$ in which no $\mathtt{X}_i$ is a subset of any $\mathtt{X}_j$. We say that such a representation is *irredundant*, and the $\mathtt{X}_i$ are the *irreducible components* of $\mathtt{X}$. Such a representation of $\mathtt{X}$ as an irredundant union of a finite number of irreducible algebraic sets is unique.

Let $\mathtt{X} \subseteq \overline{\mathbb{F}}^n$ be an irreducible algebraic closed set (variety) residing in an ambient space of dimension n. Suppose that the minimum possible number of equations required to completely describe $\mathtt{X}$ is m. Then the *dimension* of $\mathtt{X}$, denoted $\ell_{\mathtt{X}}$, is the number $(n - m)$. The varieties contained in an arbitary algebraic closed set are in general of varying dimensions. When all the varieties in a closed set have the same dimension, we will refer to it as a *uniform-dimensional* algebraic closed set.

Correspondence between rings and algebraic sets. Corresponding to the given closed set $\mathtt{X}$ there is a ring $R_{\mathtt{X}}$ obtained by quotienting the polynomial ring $\mathbb{F}_q[\bar{\mathbf{x}}]$ with the ideal generated by the polynomials which are equations of $\mathtt{X}$. That is, if $\mathtt{X}$ is the set of common zeroes of the polynomials $f_1(\bar{\mathbf{x}}), \ldots, f_m(\bar{\mathbf{x}}) \in \mathbb{F}_q[\bar{\mathbf{x}}]$ *the ring* $R_{\mathtt{X}}$ *corresponding to* $\mathtt{X}$ is

$$R_{\mathtt{X}} \overset{\text{def}}{=} \mathbb{F}_q[\bar{\mathbf{x}}]/\langle f_1(\bar{\mathbf{x}}), \cdots, f_m(\bar{\mathbf{x}})\rangle.$$

The elements of $R_{\mathtt{X}}$ can be thought of as functions from $\mathtt{X}$ to $\mathbb{F}_q$, this set of functions itself being endowed with a ring structure. The homomorphisms from $R_{\mathtt{X}}$ to $\mathbb{F}_q$ then correspond to the $\mathbb{F}_q$-rational points on $\mathtt{X}$. Indeed, $\bar{\mathbf{a}} = (a_1, \ldots, a_n) \in \mathbb{F}_q^n$ is an $\mathbb{F}_q$-rational point on $\mathtt{X}$ if and only if the map

$$\phi : R_{\mathtt{X}} \mapsto \mathbb{F}_q, \quad \phi : x_i \mapsto a_i \quad \forall 1 \le i \le n$$

is a homomorphism from $R_{\mathtt{X}}$ to $\mathbb{F}_q$.

In this way the ring $R_{\mathtt{X}}$ captures the algebraic set $\mathtt{X}$ and the structure of the ring $R_{\mathtt{X}}$ corresponds to the structure of $\mathtt{X}$. In particular, $\mathtt{X}$ is $\mathbb{F}_q$-irreducible if and only if $R_{\mathtt{X}}$ is indecomposable. $\mathtt{X}$ is absolutely irreducible (or $\overline{\mathbb{F}}_q$-irreducible) if and only if the ring $\overline{R}_{\mathtt{X}} \stackrel{\text{def}}{=} \overline{\mathbb{F}}_q[\bar{\mathbf{x}}]/\langle f_1(\bar{\mathbf{x}}), \cdots, f_m(\bar{\mathbf{x}})\rangle$ is indecomposable. In this chapter all the rings R that we will come across will be of the above form (a polynomial ring over $\mathbb{F}_q$ quotiented by some ideal $\mathcal{I}$). We will refer to the closed algebraic set corresponding to the ideal $\mathcal{I}$ as *the closed set of* R. We will denote by $R_{\mathtt{X}}$ the ring corresponding to the closed set $\mathtt{X}$ and by $\mathtt{X}_R$ the algebraic set corresponding to the ring R. We will denote by $R_{\mathtt{X}}^{fr}$ the ring of fractions of $R_{\mathtt{X}}$.

Rational maps between algebraic sets. A map of the form

$$\begin{aligned} y_1 &= \psi_1(x_1, x_2, \ldots, x_n) \\ y_2 &= \psi_2(x_1, x_2, \ldots, x_n) \\ &\vdots \\ y_m &= \psi_m(x_1, x_2, \ldots, x_n), \end{aligned}$$

where the $\psi_i = \frac{G(x_1,\ldots,x_n)}{H(x_1,\ldots,x_n)}$ are ratios of polynomials in the x_j is referred to as a *rational map.* In general, a rational map may be thought of as a function that transforms some set of points $\mathtt{X}$ in $[x_1, \ldots, x_n]$-space to a set of points $\mathtt{Y}$ in $[y_1, \ldots, y_m]$-space. Note that the denominators are polynomials and can have zeroes. Thus the map may not be defined at all points. We denote this map by $\psi : \mathtt{X} \mapsto \mathtt{Y}$. Note that for algebraic sets $\mathtt{X}$ and $\mathtt{Y}$, ψ maps points on $\mathtt{X}$ to points on $\mathtt{Y}$ if and only if the map $y_i \mapsto \psi_i(x_1, \ldots, x_n) \;\; \forall i \in [m]$ is a homomorphism from $R_{\mathtt{Y}}^{fr}$ to $R_{\mathtt{X}}^{fr}$. We will denote this ring homomorphism also by ψ itself.

A rational map $\psi : \mathtt{X} \mapsto \mathtt{Y}$ is called *birational* if it admits an inverse. That is, there exists a rational map $\phi : \mathtt{Y} \mapsto \mathtt{X}$ such that $\psi(\mathtt{X})$ has the same dimension as $\mathtt{Y}$, $\phi(\mathtt{Y})$ has the same dimension as $\mathtt{X}$, $\psi \cdot \phi = 1$ almost everywhere, and $\phi \cdot \psi = 1$ almost everywhere. In terms of the corresponding rings, it means that $(\phi \cdot \psi) : R_{\mathtt{Y}}^{fr} \mapsto R_{\mathtt{Y}}^{fr}$ is the identity map on $R_{\mathtt{Y}}^{fr}$ and $(\psi \cdot \phi) : R_{\mathtt{X}}^{fr} \mapsto R_{\mathtt{X}}^{fr}$ is the identity map on $R_{\mathtt{X}}^{fr}$.

Two algebraic closed sets $\mathtt{X}$ and $\mathtt{Y}$ are said to be *birationally equivalent* or *birational* if there exists a birational map between $\mathtt{X}$ and $\mathtt{Y}$.

A classical theorem from algebraic geometry states that *'Any algebraic variety* $\mathtt{X}$ *is birational to a hypersurface* $\mathtt{Y}$ *of the appropriate dimension'.* This theorem is a direct consequence of the well-known theorem in algebra that every finite-dimensional field extension $\mathbb{K}$ of some base field $\mathbb{F}$ is generated by some element $\gamma \in \mathbb{K}$ (i.e. $\mathbb{K} = \mathbb{F}(\gamma)$) . Moreover it can be arranged that the map $\psi : \mathtt{X} \mapsto \mathtt{Y}$ is just a linear map. That is, each unknown y_j of $\mathtt{Y}$ is expressed as a linear combination of the variables x_i of $\mathtt{X}$. The *degree* of the variety $\mathtt{X}$ is then defined to be the degree of the hypersurface $\mathtt{Y}$ birationally equivalent to $\mathtt{X}$.

6.2.1 Examples

Example: The algebraic set $\mathtt{X}$ defined by the polynomials

$$\begin{aligned} f_1(x,y,z) &= (x+y+z)(x+2y+z) \\ \text{and } f_2(x,y,z) &= (x-y)(x+y-z) \end{aligned}$$

is the irredundant union of four lines -

$$\begin{aligned} \text{line } L_1 :& (x+y+z) = (x-y) = 0, \\ \text{line } L_2 :& (x+y+z) = (x+y-z) = 0, \\ \text{line } L_3 :& (x+2y+z) = (x-y) = 0, \\ \text{and line } L_4 :& (x+2y+z) = (x+y-z) = 0. \end{aligned}$$

Generalization. In general, for polynomials $f_1(\bar{\mathbf{x}}), \ldots, f_m(\bar{\mathbf{x}}) \in \mathbb{F}[x_1, \ldots, x_n]$ where each polynomial $f_i(\bar{\mathbf{x}})$ is the product of d_i linear polynomials *in general position*, the corresponding algebraic set defined by these polynomials is the irredundant union of $(\prod_{i=1}^{m} d_i)$ hyperlines of dimension $(n-m)$.

Example: The algebraic set $\mathtt{X}$ defined by the polynomials

$$\begin{aligned} f_1(x,y,z) &= (x-y)(x+y+z)(x+2y+z) \\ \text{and } f_2(x,y,z) &= (x-y)(x+y-z) \end{aligned}$$

is the irredundant union of a plane

$$\text{plane } P_1 :(x-y) = 0$$

and two lines

$$\begin{aligned} \text{line } L_1 :& (x+y+z) = (x+y-z) = 0 \\ \text{and line } L_2 :& (x+2y+z) = (x+y-z) = 0. \end{aligned}$$

Generalization. We can generalize this example a little. Suppose that $\mathtt{X}$ is an algebraic set defined by the polynomials

$$f_1(x,y,z) = f_2(x,y,z) = 0,$$

where both f_1 and f_2 are products of linear polynomials. Moreover, suppose that $\textsc{Deg}(f_1) = d_1$, $\textsc{Deg}(f_2) = d_2$ and $\textsc{Deg}(\gcd(f_1, f_2)) = d$. Then the closed set $\mathtt{X}$ is the irredundant union of d planes and $(d_1 - d) \cdot (d_2 - d)$ lines.

Example: The algebraic closed set $\mathtt{X}$ in 3 variables x_1, x_2, x_3 defined by the equations

$$x_1^2 - x_3 = x_2^2 - (x_3 + 1) = 0$$

is an irreducible one-dimensional closed set birational to the planar curve $\mathtt{Y}$

$$y^4 - 2(2x+1)y^2 + 1 = 0$$

via the map

$$\psi : \mathtt{X} \mapsto \mathtt{Y},\ \psi : (x_1, x_2, x_3) \mapsto (x_3, x_1 + x_2)$$

The inverse map ϕ is given by

$$\phi : \mathtt{Y} \mapsto \mathtt{X},\ \phi : (x,y) \mapsto ((\frac{1}{2})(y^3 - (4x+1)y),\quad (-\frac{1}{2})(y^3 - (4x+3)y),\quad x).$$

In this example both ψ and ϕ happen to be well-defined *everywhere.*

Generalization. More generally: Suppose that $\mathtt{X}$ is an algebraic closed set in $(n+1)$ variables $x_1, x_2, \ldots, x_{n+1}$ with defining equations

$$x_1^2 - (x_{n+1} + a_1) = x_2^2 - (x_{n+1} + a_2) = \ldots = x_n^2 - (x_{n+1} + a_n) = 0.$$

Suppose further that the a_i's are all distinct. Then the closed set $\mathtt{X}$ is irreducible and birational to a planar curve of degree 2^n.

Further Generalization. Now suppose that $\mathtt{X}$ is an algebraic closed set in $(n+1)$ variables $x_1, x_2, \ldots, x_{n+1}$ with defining equations $f_1(\bar{\mathbf{x}}) = \ldots = f_n(\bar{\mathbf{x}}) = 0$ where each $f_i(\bar{\mathbf{x}})$, $1 \le i \le n$ is of the form:

$$f_i(x_1, \ldots, x_n, x_{n+1}) = \prod_{j=1}^{d} (x_i^2 - (x_{n+1} + a_{ij})).$$

Suppose further that the a_{ij}'s are all distinct. Then the closed set $\mathtt{X}$ is a union of d^n irreducible closed sets, each irreducible component being birational to a planar curve of degree 2^n.

Example: We now give an example of a *reducible* one-dimensioal closed set $\mathtt{X}$ being birational to a (reducible) planar curve. Suppose that $f_1(y), f_2(y), g_1(y), g_2(y)$ are univariate polynomials. The algebraic closed set $\mathtt{X}$ in $[x_1, x_2, y]$-space defined by the equations:

$$(x_1 - f_1(y))(x_1 - f_2(y)) = (x_2 - g_1(y))(x_2 - g_2(y)) = 0$$

is reducible and is the union of four irreducible one-dimensional closed sets. $\mathtt{X}$ is birational to the planar curve $\mathtt{Y}$ in $[z, y]$-space defined by the equation:

$$(z - f_1(y) - g_1(y))(z - f_1(y) - g_2(y))(z - f_2(y) - g_1(y))(z - f_2(y) - g_2(y)) = 0$$

via the map

$$\psi : \mathtt{X} \mapsto \mathtt{Y},\ \psi : (x_1, x_2, y) \mapsto (x_1 + x_2, y).$$

The inverse map ϕ is given by

$$\begin{aligned} &\phi : \mathtt{Y} \mapsto \mathtt{X},\ \phi : (z, y) \mapsto (B_1(z, y), B_2(z, y), y) \\ &\quad \text{with}\ \ B_1(z, y) \overset{\text{def}}{=} A_{11} f_1(y) + A_{12} f_1(y) + A_{21} f_2(y) + A_{22} f_2(y) \\ &\quad \text{and}\ \ B_2(z, y) \overset{\text{def}}{=} A_{11} g_1(y) + A_{12} g_2(y) + A_{21} g_1(y) + A_{22} g_2(y), \end{aligned}$$

where the coefficient polynomial A_{ij}'s are defined as follows. For $1 \le i, j \le 2$ define the polynomial $h_{ij}(u, y)$ as

$$h_{ij}(u, y) \overset{\text{def}}{=} \frac{g(u, y)}{(u - f_i(y) - g_j(y))}.$$

Then for $1 \le i, j \le 2$ the coefficent polynomial A_{ij} is

$$A_{ij} \overset{\text{def}}{=} \frac{h_{ij}(z, y)}{h_{ij}(f_i(y) + g_j(y), y)}.$$

6.2.2 Notation

- For an ideal $\mathcal{I} \subseteq \mathbb{F}_q[\bar{\mathbf{x}}]$, we will denote by $\textsc{Rad}(\mathcal{I})$ the radical (square-free part) of the ideal $\mathcal{I}$ defined as

$$\textsc{Rad}(\mathcal{I}) \overset{\text{def}}{=} \{f(\bar{\mathbf{x}}) \in \mathbb{F}_q[\bar{\mathbf{x}}] \mid f(\bar{\mathbf{x}})^m \in \mathcal{I} \ \text{ for some } \ m \ge 1\}.$$

- By the term *total degree* of a rational function $\psi(\bar{\mathbf{x}}) = \frac{F(\bar{\mathbf{x}})}{G(\bar{\mathbf{x}})} \in \mathbb{F}(\bar{\mathbf{x}})$, we will mean the sum of the total degrees of the numerator and the denominator. We denote it by $\textsc{Deg}(\psi)$. That is,

$$\textsc{Deg}(\psi) \overset{\text{def}}{=} \textsc{Deg}(F(\bar{\mathbf{x}})) + \textsc{Deg}(G(\bar{\mathbf{x}}))$$

6.3 Algorithm Description

6.3.1 Overview

In this section we describe in words the proposed algorithm. Our aim is to determine if a given algebraic closed set $\mathtt{X}$ over a given field $\mathbb{F}_q$ has any $\mathbb{F}_q$-rational point or not. (The set $\mathtt{X}$ is specified to us by means of polynomial equations with coefficients from $\mathbb{F}_q$). The basic idea is to decompose the given closed set $\mathtt{X}$ into a union of (possibly reducible) closed sets $\mathtt{X}_i$, each $\mathtt{X}_i$ being birational to a hypersurface $\mathtt{Y}_i$ of the appropriate dimension. We then use the partial factoring algorithm developed in the previous chapter to determine, for each i, the existence of an $\mathbb{F}_q$-rational point on the set $\mathtt{X}_i$.

We flesh out this basic idea in more detail through the rest of this section. We first describe precisely the output of the (deterministic) decomposition algorithm and show how to use our partial factoring algorithm for determining the existence of a rational point on the components of the decomposition. We then describe the decomposition algorithm itself in more detail. Finally, we remark how to improve the parallel time-complexity of the algorithm.

6.3.2 The output of the decomposition and rational points on hypersurfaces

```
Input: A finite field $\mathbb{F}_q$ and a set of polynomials $f_1, \ldots, f_m \in \mathbb{F}_q[x_1, \ldots, x_n]$ of
       total degree at most $d$.
Output: TRUE if $\exists$ an $\mathbb{F}_q$-solution to the system $f_1 = \ldots = f_m = 0$, FALSE
        otherwise.
begin
    let $c_n := 2^n$
    if $q \leq 10^5 n^3 d^{10c_n}$ then
        Check if any of the $q^n$ points in $\mathbb{F}_q^n$ is a common solution to the given
        equations and return accordingly.
    Let $\mathtt{Y}$ be the hypersurface defined by $g(y_1, \ldots, y_n) := \textsc{Rad}(f_1(y_1, \ldots, y_n))$, $\psi$ be
    the trivial map $\forall i \in [n], \psi : y_i \mapsto x_i$ and $\phi$ be its inverse. Let $\mathtt{X} \subset \overline{\mathbb{F}}_q^n$ be
    $\mathtt{X} := \langle (n-1), \mathtt{Y}, \psi, \phi \rangle$
    return Solvability($\mathtt{X}$, $f_2(\bar{\mathbf{x}}), \ldots, f_m(\bar{\mathbf{x}})$).
end
```

Algorithm 1: **SolvabilityMain :** Determine the existence of an $\mathbb{F}_q$-rational point.

```
Input: A finite field $\mathbb{F}_q$, a component $\mathtt{X} \subset \overline{\mathbb{F}}_q^n$ and a set of polynomials
       $f_1, \ldots, f_m \in \mathbb{F}_q[x_1, \ldots, x_n]$.
Output: TRUE if $\exists$ an $\mathbb{F}_q$-rational point $\bar{\mathbf{a}} \in \mathbb{F}_q^n$ which satisfies $\bar{\mathbf{a}} \in \mathtt{X}$ and
        $f_1(\bar{\mathbf{a}}) = \ldots = f_m(\bar{\mathbf{a}}) = 0$. FALSE otherwise.
begin
    Call Decompose($\mathtt{X}$, $f_1(\bar{\mathbf{x}}), \ldots, f_m(\bar{\mathbf{x}})$ ) to obtain a list $(\mathtt{X}_1, \ldots, \mathtt{X}_t)$ of
    subcomponents of $\mathtt{X}$.
    foreach component $\mathtt{X}_i := \langle \ell, \mathtt{Y}_i, \psi, \phi \rangle$ do
        Let the equation of $\mathtt{Y}_i$ be $g(y_1, \ldots, y_{\ell+1}) = 0$
        if $g(\bar{\mathbf{y}})$ has any absolutely irreducible $\mathbb{F}_q$-factor then return TRUE
        else
            for $j \leftarrow 1$ to $(\ell + 1)$ do
                Compute $h_j(\bar{\mathbf{x}}) := \psi(\frac{\partial g(\bar{\mathbf{y}})}{\partial y_j}) \in \mathbb{F}_q[x_1, \ldots, x_n]$.
            Then the closed set $\mathtt{X}'_i \subsetneq \mathtt{X}_i$,
                $$\mathtt{X}'_i \overset{\text{def}}{=} \mathtt{X}_i \cap \left( \bigcap_{j=1}^{\ell+1} \{ \bar{\mathbf{a}} \in \overline{\mathbb{F}}_q^n | h_j(\bar{\mathbf{a}}) = 0 \} \right),$$
            consists of points $\mathrm{P} \in \mathtt{X}_i$ such that $\psi(\mathrm{P}) \in \mathtt{Y}_i$ is a singular point.
            Recursively determine existence of $\mathbb{F}_q$-rational point on $\mathtt{X}'_i$ by calling
            Solvability($\mathtt{X}_i$, $h_1(\bar{\mathbf{x}}), \ldots, h_{\ell+1}(\bar{\mathbf{x}})$ )
            if $\mathtt{X}'_i$ contains a rational point then return TRUE
    return FALSE
end
```

Algorithm 2: **Solvability :** Determine the existence of an $\mathbb{F}_q$-rational point.

The number of variables is n. We will denote by c_n a constant that depends on n alone and is of size $2^{O(n)}$. Our algorithm is interesting only for large values of q; for if the size q of the given field is *small* ($q = \mathrm{O}(\mathrm{poly}(d^{c_n})\ )$), we simply do a brute force search over all possible $\mathbb{F}_q$-rational points (q^n many of them) and check if any of them belongs to $\mathtt{X}$. In what follows we shall assume that q is *large* ($q \gg d^{c_n}$).

We break the given algebraic set $\mathtt{X}$ into a union of uniform-dimensional algebraic sets $\mathtt{X}_i$: $\mathtt{X} = \bigcup \mathtt{X}_i$. These $\mathtt{X}_i$'s we call *the components of* $\mathtt{X}$. We represent a component $\mathtt{X}_i$ of $\mathtt{X}$ by a four-tuple $\langle \ell, \mathtt{Y}_i, \psi, \phi \rangle$. where:

- ℓ is the dimension of $\mathtt{X}_i$ and of $\mathtt{Y}_i$,
- $\mathtt{Y}_i$ is a hypersurface with equation $g(y_1, \ldots, y_{\ell+1}) = 0$ for some squarefree $g(\bar{\mathbf{y}}) \in \mathbb{F}_q[\bar{\mathbf{y}}]$.
- $\psi : \mathtt{X}_i \mapsto \mathtt{Y}_i$ is a rational map,
- and $\phi : \mathtt{Y}_i \mapsto \mathtt{X}_i$ is the inverse rational map of ψ.

Note that now $\mathtt{X}$ contains a $\mathbb{F}_q$-rational point if and only if some $\mathtt{X}_i$ contains a $\mathbb{F}_q$-rational point. This computation of the decomposition of $\mathtt{X}$ satisfies the following properties:

P-i). Neither $\mathtt{X}_i$ nor $\mathtt{Y}_i$ contains any singular (repeated) varieties.

P-ii). The map $\psi : \mathtt{X}_i \mapsto \mathtt{Y}_i$ is an $\mathbb{F}_q$-rational map and so is $\phi : \mathtt{Y}_i \mapsto \mathtt{X}_i$. That is the coefficients of all the rational functions occuring in ψ and ϕ are from $\mathbb{F}_q$. In particular this means that $\mathbb{F}_q$-rational points on $\mathtt{X}_i$ get mapped to $\mathbb{F}_q$-rational points on $\mathtt{Y}_i$ and vice-versa.

P-iii). The map $\psi : \mathtt{X}_i \mapsto \mathtt{Y}_i$ is well-defined on all points of $\mathtt{X}_i$. This happens because the corresponding ring homomorphism $\psi : R^{fr}_{\mathtt{Y}_i} \mapsto R^{fr}_{\mathtt{X}_i}$ is actually a linear map, mapping each generator y_i of $R_{\mathtt{Y}_i}$ to a linear combination of the generators x_j's in $R_{\mathtt{X}_i}$.

P-iv). On the other hand, the map $\phi : \mathtt{Y}_i \mapsto \mathtt{X}_i$ is well-defined everywhere except possibly at the singular points of $\mathtt{Y}_i$.

These properties ensure that if there is a $\mathbb{F}_q$-rational point on $\mathtt{X}_i$ then there is one on $\mathtt{Y}_i$ as well. In the other direction, if there is no $\mathbb{F}_q$-rational point on $\mathtt{Y}_i$ **and** there is also no singular point on $\mathtt{Y}_i$ then $\mathtt{X}_i$ does not contain any $\mathbb{F}_q$-rational point as well.

Now consider one such algebraic closed set $\mathtt{Y}_i$ of dimension ℓ. Let the equation of $\mathtt{Y}_i$ be

$$g(y_1, y_2, \ldots, y_l, y_{\ell+1}) = 0.$$

We first handle the zero-dimensional case ($\ell = 0$). In this case the components of $\mathtt{Y}_i$ are simply individual points. Moreover, by the second property, there are no singular points on $\mathtt{Y}_i$. Thus, in this case $\mathtt{X}_i$ has a rational point if and only if the univariate $g(y_1) = 0$ has an $\mathbb{F}_q$-root, or equivalently, if and only if $g(y_1)$ has an absolutely irreducible $\mathbb{F}_q$-factor (see the remark at the end of the defintion of absolute irreducibility 5.1.1).

Now consider the case when $\ell \geq 1$. We use the partial factoring algorithm described in the previous chapter to determine if $g(\bar{\mathbf{y}}) \in \mathbb{F}_q[\bar{\mathbf{y}}]$ contains any absolutely irreducible factors or not. If $g(\bar{\mathbf{y}})$ does have an absolutely irreducible $\mathbb{F}_q$-factor, then from Weil's theorem we can deduce that there does exist an $\mathbb{F}_q$-rational point on $\mathtt{Y}_i$. Indeed, Weil's theorem says that any absolutely irreducible polynomial contains *a lot of* ($\Theta(q^\ell)$, provided q is large enough in comparison to the degree of the polynomial) rational points. Thus if $g(\bar{\mathbf{y}})$ has an absolutely irreducible $\mathbb{F}_q$-factor $g_1(\bar{\mathbf{y}})$ then the hypersurface $g_1(\bar{\mathbf{y}}) = 0$ has *a lot of* $\mathbb{F}_q$-rational points. Moreover, most of these points are non-singular. There is also a partial converse to Weil's theorem: if $g(\bar{\mathbf{y}}) = 0$ has no absolutely irreducible factors then any rational point on $g(\bar{\mathbf{y}}) = 0$, if it exists, is a singular point.

Thus if $g(\bar{\mathbf{y}})$ has an absolutely irreducible factor we deduce that $\mathtt{X}_i$, and hence $\mathtt{X}$, contains a $\mathbb{F}_q$-rational point and we stop. Otherwise any rational point on $\mathtt{X}_i$, if it exists, must map to a singular point on $\mathtt{Y}_i$ under ψ. Now the set of points on $\mathtt{X}_i$ that can map to a singular point on $\mathtt{Y}_i$ under ψ is a closed algebraic subset of $\mathtt{X}_i$ of dimension strictly less than ℓ. We compute the equations of this subset and then repeat the process to determine if this smaller dimensional set has a rational point or not. This process continues until the $\mathtt{X}_i$'s that we get are zero-dimensional.

6.3.3 Description of the decomposition algorithm.

The most general form of the algebraic set decomposition problem is the following -

> **Algebraic Set Decomposition Problem.** Consider a set of polynomials $f_1(\bar{\mathbf{x}}), \ldots, f_m(\bar{\mathbf{x}}) \in \mathbb{F}_q[\bar{\mathbf{x}}]$ of total degree d in n variables over the finite field $\mathbb{F}_q$. Decompose the algebraic set defined by
>
> $$f_1(\bar{\mathbf{x}}) = f_2(\bar{\mathbf{x}}) = \ldots = f_m(\bar{\mathbf{x}}) = 0$$

into $\mathbb{F}_q$-irreducible components, representing each of them by a birational hypersurface over $\mathbb{F}_q$, together with a map from the component to the hypersurface and an inverse rational map from the hypersurface to the component.

Lacking an efficient algorithm for completely factoring univariate polynomials, we cannot solve this most general form of the decomposition problem. We do solve this problem partially and as we shall see, this partial solution is good enough for deciding solvability.

```
Input: A finite field $\mathbb{F}_q$, an algebraic closed set $\mathtt{X}$ over $\mathbb{F}_q$, and polynomials
       $f_1, \ldots, f_m \in \mathbb{F}_q[x_1, \ldots, x_n]$ .
Output: A list $\langle \mathtt{X}_1, \ldots, \mathtt{X}_t \rangle$ of (possibly reducible) components of the closed set
        $\mathtt{X} \cap \left( \bigcap_{i=1}^{m} \{ \bar{\mathbf{a}} \in \overline{\mathbb{F}}_q^n | f_i(\bar{\mathbf{a}}) = 0 \} \right)$.
begin
    Initialize a list $L$ with the single component $\mathtt{X}$.
    for $i \leftarrow 1$ to $m$ do
        Initialize $L'$ to be the empty list.
        forall $\widehat{\mathtt{X}} := \langle \ell, \mathtt{Y}, \psi, \phi \rangle$ in the list $L$ do
            Let $\mathbb{F} \overset{\mathrm{def}}{=} \mathbb{F}_q(y_1, \ldots, y_\ell)$,  $R_{\mathtt{Y}}^{fr} := \mathbb{F}[y_{\ell+1}] / \langle g(y_1, \ldots, y_{\ell+1}) \rangle$,
            $f_i^{\phi}(y_1, \ldots, y_{\ell+1}) := \phi(f_i(x_1, \ldots, x_n)) = \dfrac{h_1(y_1, \ldots, y_{\ell+1})}{h_2(y_1, \ldots, y_\ell)} \in R_{\mathtt{Y}}^{fr}$
            where $h_1$ and $h_2$ are polynomials in the $y_j$'s.
            We now have two hypersurfaces $g(\bar{\mathbf{y}}) = 0$ and $h_1(\bar{\mathbf{y}}) = 0$ in the ambient
            $[y_1, \ldots, y_{\ell+1}]$-space. Compute the intersection of these two hypersurfaces
            and obtain two components $\widehat{\mathtt{Y}}_1 := \langle \ell, \mathtt{Z}_1, \psi_1, \phi_1 \rangle$ and
            $\widehat{\mathtt{Y}}_2 := \langle \ell - 1, \mathtt{Z}_2, \psi_2, \phi_2 \rangle$.
            if $\widehat{\mathtt{Y}}_1 \neq \emptyset$ then
                add the component $\widehat{\mathtt{X}}_1 := \langle \ell, \mathtt{Z}_1, \psi_1 \circ \psi, \phi \circ \phi_1 \rangle$ to $L'$.
            if $\widehat{\mathtt{Y}}_2 \neq \emptyset$ then
                add the component $\widehat{\mathtt{X}}_2 := \langle \ell - 1, \mathtt{Z}_2, \psi_2 \circ \psi, \phi \circ \phi_2 \rangle$ to $L'$.
        $L \leftarrow L'$
    Output the list $L$
end
```

Algorithm 3: **Decompose -** Compute the decomposition of an algebraic set.

```
Input: A finite field $\mathbb{F}_q$ and two $\ell$-dimensional hypersurfaces
       $\mathtt{Y}_1 : g_1(y_1, \ldots, y_{\ell+1}) = 0$ and $\mathtt{Y}_2 : g_2(y_1, \ldots, y_{\ell+1}) = 0$.
Output: The decomposition of $(\mathtt{Y}_1 \cap \mathtt{Y}_2)$ as the union of two component closed
        subsets $\widehat{\mathtt{Y}}_1 := \langle \ell, \mathtt{Z}_1, \psi_1, \phi_1 \rangle$ and $\widehat{\mathtt{Y}}_2 := \langle \ell - 1, \mathtt{Z}_2, \psi_2, \phi_2 \rangle$.
begin
    $g_1(\bar{\mathbf{y}}) \leftarrow \textsc{Rad}(g_1(\bar{\mathbf{y}})), \quad g_2(\bar{\mathbf{y}}) \leftarrow \textsc{Rad}(g_2(\bar{\mathbf{y}}))$
    By making a suitable linear transformation $\sigma$ on the variables $y_1, \ldots y_{\ell+1}$,
    ensure that both $\sigma(g_1(\bar{\mathbf{y}}))$ and $\sigma(g_2(\bar{\mathbf{y}}))$ are monic and seperable polynomials
    with respect to $y_{\ell+1}$.
    $$\text{Let} \quad \mathbb{F} \overset{\text{def}}{=} \mathbb{F}_q(y_1, \ldots, y_{\ell-1}), \quad R := \mathbb{F}(y_\ell)[y_{\ell+1}]/\langle \sigma(g_1(\bar{\mathbf{y}})), \sigma(g_2(\bar{\mathbf{y}})) \rangle.$$
    Compute
    $$h(\bar{\mathbf{y}}) := \gcd(\sigma(g_1(\bar{\mathbf{y}})), \sigma(g_2(\bar{\mathbf{y}}))), h_1(\bar{\mathbf{y}}) := \frac{\sigma(g_1(\bar{\mathbf{y}}))}{h(\bar{\mathbf{y}})}, h_2(\bar{\mathbf{y}}) := \frac{\sigma(g_2(\bar{\mathbf{y}}))}{h(\bar{\mathbf{y}})}.$$
    Note that $h(\bar{\mathbf{y}}), h_1(\bar{\mathbf{y}}), h_2(\bar{\mathbf{y}}) \in \mathbb{F}_q[\bar{\mathbf{y}}]$ are all monic polynomials in $y_{\ell+1}$. The ring
    $R$ then decomposes into the direct sum of two rings:
    $$R = \left(R_1 \overset{\text{def}}{=} \mathbb{F}(y_\ell)[y_{\ell+1}]/\langle h(\bar{\mathbf{y}}) \rangle\right) \oplus \left(R_2 \overset{\text{def}}{=} \mathbb{F}[y_\ell, y_{\ell+1}]/\textsc{Rad}(\langle h_1(\bar{\mathbf{y}}), h_2(\bar{\mathbf{y}}) \rangle)\right)$$
    Let $\pi_1 : R \mapsto R_1$ and $\pi_2 : R \mapsto R_2$ be the projection maps. Also let $\rho_1 : R_1 \mapsto R$
    and $\rho_2 : R_2 \mapsto R$ be the natural inclusion maps.
    if $\textsc{Deg}(h(\bar{\mathbf{y}})) = 0$ then $\widehat{\mathtt{Y}}_1 \overset{\text{def}}{=} \emptyset$ else
        $\widehat{\mathtt{Y}}_1 \overset{\text{def}}{=} \langle \ell, \mathtt{Z}_1 := \{\bar{\mathbf{a}} \in \overline{\mathbb{F}}_q^{\ell+1} \mid h(\bar{\mathbf{a}}) = 0\}, \sigma^{-1} \cdot \rho_1, \pi_1 \cdot \sigma \rangle$
    Viewing $R_2$ as an algebra over $\mathbb{F}$, use the primitive element theorem to obtain a
    ring $R_{\mathtt{Z}}^{fr} := \mathbb{F}[z]/\langle \tilde{g}(z) \rangle$ such that $\phi : R_2 \mapsto R_{\mathtt{Z}}^{fr}$ is an isomorphism with inverse
    $\psi$. Here $\mathtt{Z} := \{\bar{\mathbf{a}} \in \overline{\mathbb{F}}_q^{\ell} \mid \tilde{g}(\bar{\mathbf{a}}) = 0\}$ is the algebraic closed set corresponding to
    $R_{\mathtt{Z}}^{fr}$.
    if $\mathtt{Z} = \emptyset$ then $\widehat{\mathtt{Y}}_2 = \emptyset$ else $\widehat{\mathtt{Y}}_2 \overset{\text{def}}{=} \langle \ell - 1, \mathtt{Z}, \sigma^{-1} \cdot \rho_2 \cdot \psi, \phi \cdot \pi_2 \cdot \sigma \rangle$
    return $\langle \widehat{\mathtt{Y}}_1, \widehat{\mathtt{Y}}_2 \rangle$.
end
```

Function `Intersect` - Compute the intersection of two hypersurfaces.

We now delve a little deeper and describe in more detail the process of computing the components together with their birationally equivalent hypersurfaces.

Let $\mathtt{X}^{[i]}$ be the closed set defined by the first i equations:

$$f_1(\bar{\mathbf{x}}) = f_2(\bar{\mathbf{x}}) = \ldots = f_i(\bar{\mathbf{x}}) = 0.$$

Corresponding to the closed set $\mathtt{X}^{[i]}$ we have the ring

$$R_{\mathtt{X}}^{[i]} := \mathbb{F}_q[\bar{\mathbf{x}}]/\langle f_1(\bar{\mathbf{x}}), \ldots, f_i(\bar{\mathbf{x}})\rangle.$$

Starting with $i = 1$, our algorithm successively computes the decomposition of $\mathtt{X}^{[i]}$ for $i = 2, 3, \ldots, m$ until we get the decomposition of $\mathtt{X}^{[m]} = \mathtt{X}$. Our algorithm ensures that at each stage the components that we get are all '*square-free*', i.e. each variety in the component occurs with multiplicity 1.

In order to get the decomposition of the closed set $\mathtt{X}^{[i+1]}$ from that of $\mathtt{X}^{[i]}$, we compute the intersection of each component of $\mathtt{X}^{[i]}$ with the hypersurface $\mathtt{Z}$ defined by $f_{i+1}(\bar{\mathbf{x}}) = 0$. Consider one such component $\widehat{\mathtt{X}}$ of $\mathtt{X}^{[i]}$, of dimension ℓ. Then $\widehat{\mathtt{X}} \cap \mathtt{Z}$ is the union of two components $\widehat{\mathtt{X}}_1$ and $\widehat{\mathtt{X}}_2$. $\widehat{\mathtt{X}}_1$ is the union of those ℓ-dimensional varieties in $\widehat{\mathtt{X}}$ that are a subset of $\mathtt{Z}$. Each of the remaining varieties in $\widehat{\mathtt{X}} - \widehat{\mathtt{X}}_1$ give a collection of $(\ell-1)$-dimensional varieties upon intersection with $\mathtt{Z}$, the union of which is the set $\widehat{\mathtt{X}}_2$. In this way intersecting a component of $\mathtt{X}^{[i]}$ with the hypersurface $f_{i+1}(\bar{\mathbf{x}}) = 0$ gives, in general, two components of $\mathtt{X}^{[i+1]}$. Continuing in this manner we get the decomposition of $\mathtt{X} = \mathtt{X}^{[m]}$. It remains for us to describe how to compute the intersection of a component with a hypersurface.

Computing the intersection of a component $\widehat{\mathtt{X}} := \langle \ell, \mathtt{Y}, \psi, \phi\rangle$ with a hypersurface $f_i(\bar{\mathbf{x}}) = 0$. The component $\widehat{\mathtt{X}}$ is birational to a ℓ-dimensional hypersurface $\mathtt{Y}$ with defining equation $g(y_1, \ldots, y_{\ell+1}) = 0$. We 'project' the constraint $f_i(\bar{\mathbf{x}}) = 0$ into the ambient $[y_1, \ldots, y_{\ell+1}]$-space of $\mathtt{Y}$ by using the map $\phi : R_{\widehat{\mathtt{X}}}^{fr} \mapsto R_{\mathtt{Y}}^{fr}$. Thus the problem now boils down to computing the intersection of two hypersurfaces $g_1(\bar{\mathbf{y}}) := g(\bar{\mathbf{y}})$ and $g_2(\bar{\mathbf{y}}) := \phi(f_i(\bar{\mathbf{x}}))$. After some initial preprocessing, we compute $h(\bar{\mathbf{y}}) = \gcd(g_1(\bar{\mathbf{y}}), g_2(\bar{\mathbf{y}}))$ and this captures all the varieties common to both $g_1(\bar{\mathbf{y}}) = 0$ and $g_2(\bar{\mathbf{y}}) = 0$. The hypersurface $h(\bar{\mathbf{y}}) = 0$ then gives us the representation of $\widehat{\mathtt{X}}_1$. After removing these common varieties from both $g_1(\bar{\mathbf{y}}) = 0$ and $g_2(\bar{\mathbf{y}}) = 0$, our problem boils down to computing an $(\ell - 1)$-dimensional hypersurface birational to the intersection of two 'disjoint' ℓ-dimensional hypersurfaces. We solve this problem by using the primitive element theorem as described in the next section and upon composing the relevant maps we obtain a hypersurface-representation of $\widehat{\mathtt{X}}_2$ as well.

In summary, we obtain the decomposition of the given set by introducing the constraints one by one and at each stage computing the intersection of every component with the newly introduced constraint. This completes the description of the sequential version of our algorithm.

▮ *Time complexity of the algorithm.*

The computation of the decomposition of the given algebraic set $\mathtt{X}$ can be viewed in terms of a binary tree of depth m where the nodes at depth i correspond to the components in the decomposition of the closed set $\mathtt{X}^{[i]}$. We will observe that the degree of any hypersurface is bounded by d^{c_n}. Also, the total degree of every rational function that occurs in the map from the given set $\mathtt{X}$ to the hypersurfaces that occur during the computation process is also bounded by d^{c_n}. From this it follows that the total number of $\mathbb{F}_q$-field operations that we require is $\mathrm{poly}(d^{c_n} \cdot k_m)$ where k_m is the number of components output by the decomposition algorithm. Finally, k_m is itself upper-bounded by d^{c_n} thereby implying an overall time complexity of $\mathrm{poly}(d^{c_n} \cdot m)$ field operations over $\mathbb{F}_q$. Note that both the degree and the number of components of $\mathtt{X}$ are bounded by d^{c_n}, a quantity that, remarkably, is independent of m.

▮ *Parallelizing the algorithm.*

Consider once again the binary tree corresponding to the computation of the decomposition algorithm as mentioned in the previous section.

The fundamental operations involved in the decomposition algorithm are computing the gcd of two polynomials, solving a set of linear equations and computing the characteristic polynomial of a matrix. All of these are all well-studied operations known to be efficiently parallelizable. Thus, by doing an efficient parallel implementation of these fundamental operations and a parallel traversal of the aforementioned computation tree, we get a parallel time complexity of $poly(c_n \cdot \log d \cdot m \cdot \log q)$. To make the dependence polylogarithmic in m also we need one more idea. The idea is simply to divide the given set of m equations into two sets of size $\frac{m}{2}$, compute the decomposition of the closed algebraic set induced by each set of equations recursively in parallel and then take the intersection of each pair of components to get the decomposition of the original algebraic set $\mathtt{X}$. Let $\widehat{\mathtt{X}}$ be the algebraic closed set corresponding to the equations

$$f_1(\bar{\mathbf{x}}) = f_2(\bar{\mathbf{x}}) = \ldots = f_{\frac{m}{2}}(\bar{\mathbf{x}}) = 0$$

and $\widetilde{\mathtt{X}}$ be the algebraic closed set corresponding to the rest of the equations

$$f_{\frac{m}{2}+1}(\bar{\mathbf{x}}) = f_{\frac{m}{2}+1}(\bar{\mathbf{x}}) = \ldots = f_m(\bar{\mathbf{x}}) = 0.$$

We recursively compute the decomposition of $\widehat{\mathtt{X}}$ and $\widetilde{\mathtt{X}}$ in parallel. Let $\widehat{\mathtt{X}} = \bigcup_i \widehat{\mathtt{X}}_i$ and $\widetilde{\mathtt{X}} = \bigcup_j \widetilde{\mathtt{X}}_j$ be the decomposition of $\widehat{\mathtt{X}}$ and $\widetilde{\mathtt{X}}$ respectively. Then the decomposition of $\mathtt{X}$ is given simply by $\mathtt{X} = \bigcup_{i,j}(\widehat{\mathtt{X}}_i \cap \widetilde{\mathtt{X}}_j)$. The intersection of every pair of sets $\widehat{\mathtt{X}}_i$ and $\widetilde{\mathtt{X}}_j$ is computed again in parallel and computing one such intersection again involves elementary linear algebraic operations which are also efficiently parallelized. Overall, this gives a parallel time complexity of $poly(c_n \cdot \log d \cdot \log m \cdot \log q)$.

6.3.4 The Primitive Element Theorem

We now come to the main technical section of our algorithm - computing the intersection of two hypersurfaces. In this subsection we give a very constructive version of the well known primitive element theorem (cf. Lang [Lan94]), along with explicit bounds on the sizes of the involved quantities, as required for our purposes.

Consider polynomials

$$f_1(z_1, \ldots, z_n, x) \in \mathbb{F}_q[z_1, \ldots, z_n, x, y] \quad \text{and} \quad f_2(z_1, \ldots, z_n, y) \in \mathbb{F}_q[z_1, \ldots, z_n, x, y].$$

Let $f_1(\bar{\mathbf{z}}, x)$ and $f_2(\bar{\mathbf{z}}, y)$ be squarefree polynomials of total degree d_1 and d_2 respectively over $\mathbb{F}_q$. Moreover, suppose that $f_1(\bar{\mathbf{z}}, x)$ is monic and separable with respect to the variable x while $f_2(\bar{\mathbf{z}}, y)$ is monic and separable with respect to the variable y.

Remark. If f_1 and f_2 are not monic and separable then a random linear transformation σ on the variables makes them monic and separable so that in this case we apply the appropriate linear transform on the variables and work with these new polynomials instead. See [Kal82] for a proof of the bivariate case. The proof of the general case in n variables is an easy generalization of the bivariate case. Moreover, when the number of variables is bounded such a transformation σ can be computed efficiently [Kal82].

Let $\mathbb{F}$ be the rational function field $\mathbb{F} \overset{\text{def}}{=} \mathbb{F}_q(z_1, \ldots, z_n)$. Let R be the ring $\mathbb{F}[x, y]/\langle f_1(x), f_2(y)\rangle$. Thus R is an algebra of dimension $d_1 \cdot d_2$ over the field $\mathbb{F}$ with basis

$$\mathcal{B}_1 \overset{\text{def}}{=} \{x^i y^j \mid 0 \leq i < d_1, 0 \leq j < d_2\}.$$

We want to express R as a ring of the form $\mathbb{F}[z]/\langle g(z)\rangle$. We will see that choosing $g(z)$ to be the minimal polynomial of some element $\alpha \in R$ of the form $\alpha = x + ty$ with $t \in \mathbb{F}_q$ works for us.

Suppose that in the algebraic closure $\overline{\mathbb{F}}$ of $\mathbb{F}$, f_1 and f_2 factor as:

$$f_1(x) = \prod_{i=1}^{d_1}(x - \alpha_i),$$

$$f_2(y) = \prod_{j=1}^{d_2}(x - \beta_j).$$

By the squarefreeness and separability of f_1 the α_i's are all distinct. Similarly the β_j's are all distinct.

Now for some $t \in \mathbb{F}_q$, consider the element $\alpha \in R$ defined as $\alpha \overset{\text{def}}{=} (x + ty)$. Then the characteristic polynomial of α over $\mathbb{F}$ is

$$g(z) \overset{\text{def}}{=} \text{charpoly}_{\alpha/\mathbb{F}}(z) = \prod_{i=1}^{d_1}\prod_{j=1}^{d_2}(z - (\alpha_i + t\beta_j)).$$

Let $A \subset \overline{\mathbb{F}}$ be the set

$$A \overset{\text{def}}{=} \{(\alpha_{i_1} - \alpha_{i_2})/(\beta_{j_1} - \beta_{j_2}) \ \mid i_1, i_2 \in [d_1], j_1 \neq j_2 \in [d_2]\}.$$

Then for $t \notin A$, the roots of $g(z)$ are all distinct. Fix any such $t \notin A$. Then since the characteristic polynomial $g(z)$ of α is squarefree and separable, it is in fact also the minimal polynomial of α. Therefore $R = \mathbb{F}(\alpha) = \mathbb{F}[z]/\langle g(z)\rangle$. Choosing any $t \in (\mathbb{F}_q \setminus A)$ gives a suitable α. Note that $\mid A \mid < d_1^2 d_2^2$ and therefore there are at least $\mid (\mathbb{F}_q \setminus A) \mid \geq (q - d_1^2 d_2^2)$ suitable choices of t.

We now adopt a slightly different viewpoint of the above matter. The discussion above explicitly exhibits an isomorphism ψ from the ring $R_1 \overset{\text{def}}{=} \mathbb{F}[z]/\langle g(z)\rangle$ to the ring $R \overset{\text{def}}{=} \mathbb{F}[x, y]/\langle f_1(x), f_2(y)\rangle$ given by $\psi : z \mapsto (x+ty)$, where $g(\bar{\mathbf{z}}, z) \in \mathbb{F}_q[\bar{\mathbf{z}}, z]$ is the minpoly of the element $(x + ty) \in R$. Let $\phi : R \mapsto R_1$ be the inverse of ψ. Clearly then ϕ can be viewed as a map from the set of points $\mathtt{Y}$ on $g(\bar{\mathbf{z}}, z) = 0$ to the set of points $\mathtt{X}$ on $f_1(\bar{\mathbf{z}}, x) = f_2(\bar{\mathbf{z}}, y) = 0$. ψ then maps the points on $\mathtt{X}$ to points on $\mathtt{Y}$ and by the linear nature of the map, ψ is well-defined everywhere.

We now investigate the well-definedness of ϕ as a map from points in $\mathtt{Y}$ to points in $\mathtt{X}$. For $P = (\bar{\mathbf{z}}, z)$ let $\phi(P) = (\bar{\mathbf{z}}, \phi_1(P), \phi_2(P))$. Over the algebraic closure $\overline{\mathbb{F}}$ of $\mathbb{F}$, we can obtain an explicit expression for ϕ_1 as a polynomial in z. Indeed this expression is remniscient of polynomial interpolation for the following reason. If $x = \phi_1(z) \in \overline{\mathbb{F}}[z]$ is the expression for x in terms of z then we want it to satisfy $\phi_1(\alpha_i + t\beta_j) = \alpha_i$ for all $i \in [d_1]$ and $j \in [d_2]$. Let $g_{ij}(z) \overset{\text{def}}{=} \frac{g(z)}{z-(\alpha_i+t\beta_j)} \in \overline{\mathbb{F}}[z]$. Its easy to verify that

$$\phi_1(z) := \sum_{i,j} \frac{g_{ij}(z)}{g_{ij}(\alpha_i + t\beta_j)} \alpha_i$$

works. It turns out the rhs of the above equation is actually in $\mathbb{F}[z]$ itself. From the above expression, we can deduce that $\phi_1(P)$ is well defined for all non-singular points P on $\mathtt{Y}$. Similarly, it can be shown that $\phi_2(P)$ is also well-defined for all non-singular points P on $\mathtt{Y}$.

Let us summarize the above discussion far as a theorem.

Proposition 6.3.1. *(***Primitive Element Theorem.***) Let $\mathbb{F}_q$ be a finite field. Let $f_1(z_1, \ldots, z_n, x) \in \mathbb{F}_q[z_1, \ldots, z_n, x, y]$ and $f_2(z_1, \ldots, z_n, y) \in \mathbb{F}_q[z_1, \ldots, z_n, x, y]$ be square-free polynomials of degree d_1 and d_2 respectively over $\mathbb{F}_q$. Moreover, $f_1(\bar{\mathbf{z}}, x)$ is monic and separable with respect to the variable x while $f_2(\bar{\mathbf{z}}, y)$ is monic and separable with respect to the variable y. Let $\mathbb{F}$ be the rational function field $\mathbb{F} \overset{\text{def}}{=} \mathbb{F}_q(z_1, \ldots, z_n)$. Let R be the ring $\mathbb{F}[x, y]/\langle f_1(x), f_2(y)\rangle$. Thus R is an algebra of dimension $d_1 \cdot d_2$ over the field $\mathbb{F}$. Then R is isomophic to the ring $R_1 := \mathbb{F}[z]/\langle g(z)\rangle$, where $g(z) \in \mathbb{F}_q[z_1, \ldots, z_n, z]$ is a polynomial of degree $(d_1 \cdot d_2)$ and is monic in z. The map $\psi : R_1 \mapsto R, \quad \psi : z \mapsto (x + ty)$ for some $t \in \mathbb{F}_q$ is a ring isomorphism. Let $\phi : R \mapsto R_1$ be the inverse of ψ. Then ϕ maps points on the closed set of $g(\bar{\mathbf{z}}, z) = 0$ to points on the closed set of $f_1(\bar{\mathbf{z}}, x) = f_2(\bar{\mathbf{z}}, y) = 0$ in such a way that it is well-defined on all non-singular points on $g(\bar{\mathbf{z}}, z) = 0$.*

Moreover the ring R_1 together with the maps ψ and ϕ can be constructed in deterministic polynomial time (i.e. time polynomial in the size of the input and output).

6.3.5 Intersection of two hypersurfaces.

Now suppose that we are given two $(n + 1)$-dimensional hypersurfaces

$$f_1(z_1, \ldots, z_n, x, y) = 0 \quad \text{and} \quad f_2(z_1, \ldots, z_n, x, y) = 0$$

over the field $\mathbb{F}_q$. Moreover assume that f_1 and f_2 have no common varieties, i.e. the polynomials $f_1(\bar{\mathbf{z}}, x, y)$ and $f_2(\bar{\mathbf{z}}, x, y)$ are coprime. We want to compute an n-dimensional hypersurface $g(\bar{\mathbf{z}}, z) = 0$ birational to their intersection

$$f_1(\bar{\mathbf{z}}, x, y) = f_2(\bar{\mathbf{z}}, x, y) = 0.$$

Equivalently, we want to compute a ring R_1 of the form

$$R_1 = \mathbb{F}_q(\bar{\mathbf{z}})[z]/\langle g(z)\rangle$$

that is $\mathbb{F}_q(\bar{\mathbf{z}})$-isomorphic to the given ring

$$R = \mathbb{F}_q(\bar{\mathbf{z}})[x,y]/\textsc{Rad}(\langle f_1(\bar{\mathbf{z}},x,y), f_2(\bar{\mathbf{z}},x,y)\rangle).$$

We do this as follows:

1. Compute

$$\begin{aligned} h_1(\bar{\mathbf{z}},x) &= \textsc{Rad}(\textsc{Resultant}_y(f_1(\bar{\mathbf{z}},x,y), f_2(\bar{\mathbf{z}},x,y))) \neq 0 \\ \text{and}\ \ h_2(\bar{\mathbf{z}},y) &= \textsc{Rad}(\textsc{Resultant}_x(f_1(\bar{\mathbf{z}},x,y), f_2(\bar{\mathbf{z}},x,y))) \neq 0. \end{aligned}$$

   Then over the field $\mathbb{F} = \mathbb{F}_q(\bar{\mathbf{z}})$, since $h_1(x)$ and $h_2(y) \in \langle f_1(x,y), f_2(x,y)\rangle$, we have

$$R = (\mathbb{F}[x,y]/\langle h_1(x), h_2(y)\rangle)/\langle f_1(x,y), f_2(x,y)\rangle.$$

2. Let $S \overset{\text{def}}{=} \mathbb{F}[x,y]/\langle h_1(x), h_2(y)\rangle$. Using the primitive element theorem described previously obtain a ring S' of the form $S' = \mathbb{F}[z]/\langle g_1(z)\rangle$ along with isomorphisms $\phi : S \mapsto S'$ and $\psi : S' \mapsto S$.

3. Viewing $f_1(x,y)$ and $f_2(x,y)$ as elements of S, compute

$$f_1'(z) = \phi(f_1(x,y)) \in S', \quad f_2'(z) = \phi(f_2(x,y)) \in S'.$$

   Then $R \subseteq S$ is isomorphic to $S'/\langle f_1'(z), f_2'(z)\rangle = \mathbb{F}[z]/\langle g(z)\rangle$ where $g(z) = gcd(g_1(z), f_1'(z), f_2'(z))$. The restriction of the map ϕ to $R \subseteq S$ provides the isomorphism from R to $R_1 := \mathbb{F}[z]/\langle g(z)\rangle \subseteq S'$.

Clearly all these computations are in deterministic polynomial time. Finally, when ψ and ϕ are viewed as mappings from one algebraic closed set to another, ψ is well defined at all points whereas ϕ is well-defined at all non-singular points. We summarize this as a theorem.

Proposition 6.3.2. *Let $\mathbb{F}_q$ be a finite field. Let $f_1(z_1,\ldots,z_n,x,y) \in \mathbb{F}_q[z_1,\ldots,z_n,x,y]$ and $f_2(z_1,\ldots,z_n,x,y) \in \mathbb{F}_q[z_1,\ldots,z_n,x,y]$ be squarefree polynomials of degree d_1 and d_2 respectively over $\mathbb{F}_q$. Let $\mathbb{F}$ be the rational function field $\mathbb{F} \overset{\text{def}}{=} \mathbb{F}_q(z_1,\ldots,z_n)$. Let R be* *the ring $\mathbb{F}[x,y]/\langle f_1(x), f_2(y)\rangle$. Then R is isomophic to the ring $R_1 := \mathbb{F}[z]/\langle g(z)\rangle$, where $g(z) \in \mathbb{F}_q[z_1, \ldots, z_n, z]$. The map $\psi : R_1 \mapsto R, \quad \psi : z \mapsto (x + ty)$ for some $t \in \mathbb{F}_q$ is a ring isomorphism. Let $\phi : R \mapsto R_1$ be the inverse of ψ. Then ϕ maps points on the closed set of $g(\bar{\mathbf{z}}, z) = 0$ to points on the closed set of $f_1(\bar{\mathbf{z}}, x) = f_2(\bar{\mathbf{z}}, y) = 0$ in such a way that it is well-defined on all non-singular points on $g(\bar{\mathbf{z}}, z) = 0$.*

Moreover the ring R_1 together with the maps ψ and ϕ can be constructed in deterministic polynomial time (i.e. time polynomial in the size of the input and output).

6.3.6 Proof of Correctness

We now prove the correctness of our algorithm. The main subroutine involved in the decomposition is computing the intersection of two hypersurfaces. The important properties of this intersection algorithm and its proof of correctness has already been discussed. So we can assume that the decomposition algorithm works correctly and returns a list of components of the given algebraic set. Now consider a uniform-dimensional component $\mathtt{X}_i := \langle \ell, \mathtt{Y}_i, \psi, \phi\rangle$ in the list of components returned by the decomposition algorithm. Our algorithm then consists of two cases.

Case I: $\mathtt{Y}_i$ contains an absolutely irreducible hypersurface. We will make use of the following two results by Schmidt [Sch74].

Theorem 6.3.3. *Suppose $g(y_1, \ldots, y_{\ell+1})$ is an absolutely irreducible polynomial of total degree $d > 0$, with coefficients in the finite field $\mathbb{F}_q$. Let A be the number of $\mathbb{F}_q$-rational points on*

$$g(y_1, \ldots, y_{\ell+1}) = 0.$$

Suppose

$$q > 10^4 \ell^3 d^5 P^3(4 \lfloor logd \rfloor),$$

where $P(1) = 2, P(2) = 3, \ldots$ is the sequence of primes. In particular $P(x) \approx x \log x$, and hence the right hand side of the above inequality is $O(\ell^3 d^{5+\epsilon})$ for every $\epsilon > 0$. Then

$$A > q^\ell - (d)(d-1)q^{\ell-(1/2)}.$$

Theorem 6.3.4. *Suppose $g_1(y_1, \ldots, y_{\ell+1}), \ldots, g_m(y_1, \ldots, y_{\ell+1})$ are polynomials of degree $\leq d$ with coefficients in $\mathbb{F}_q$ and without a common factor. Then the number of $\mathbb{F}_q$-rational points on*

$$g_1(y_1, \ldots, y_{\ell+1}) = \ldots = g_m(y_1, \ldots, y_{\ell+1}) = 0$$

is $\leq 2\ell d^3 q^{\ell-1}$.

Combining these two theorems we prove the following:

Theorem 6.3.5. *Suppose that* $g(y_1, \ldots, y_{\ell+1}) \in \mathbb{F}_q[y_1, \ldots, y_{\ell+1}]$ *is a squarefree polynomial of total degree* d *having at least one absolutely irreducible* $\mathbb{F}_q$*-factor. If* $q \geq 10^5 \ell^3 d^{10}$*, then there exists at least one* non-singular $\mathbb{F}_q$-rational point *on the hypersurface*

$$g(y_1, \ldots, y_{\ell+1}) = 0.$$

Proof. If $d = 1$ then $g(\bar{\mathbf{y}})$ is simply a hyperplane of dimension (ℓ) and thus all the q^ℓ $\mathbb{F}_q$-rational points on $g(\bar{\mathbf{y}})$ are non-singular. So now assume $d \geq 2$.

Since $g(\bar{\mathbf{y}})$ is squarefree, therefore it must be coprime to at least one of its partial derivatives $\left(\frac{\partial g}{\partial y_i}\right)(\bar{\mathbf{y}})$. So by theorem 6.3.4 the system of equations

$$g(\bar{\mathbf{y}}) = \left(\frac{\partial g}{\partial y_1}\right)(\bar{\mathbf{y}}) = \ldots = \left(\frac{\partial g}{\partial y_{\ell+1}}\right)(\bar{\mathbf{y}}) = 0$$

has at most $2\ell d^3 q^{\ell-1}$ solutions. In other words the number of $\mathbb{F}_q$-rational singular points on $g(\bar{\mathbf{y}})$ is upper bounded by $2\ell d^3 q^{\ell-1}$.

Let $g_1(\bar{\mathbf{y}}) \in \mathbb{F}_q[\bar{\mathbf{y}}]$ be an absolutely irreducible $\mathbb{F}_q$-factor of $g(\bar{\mathbf{y}})$. Combining the lower bound of 6.3.3 on the number of $\mathbb{F}_q$-rational points on $g_1(\bar{\mathbf{y}}) = 0$ with this upper bound on the number of singular points on $g(\bar{\mathbf{y}}) = 0$, we get that there exists at least one non-singular $\mathbb{F}_q$-rational point on $g(\bar{\mathbf{y}}) = 0$.

□

We need to bound the number and degree of the components of various dimensions obtained as our algorithm s. We bound it as follows.

Lemma 6.3.6. *During the execution of the algorithm, the degree of any* ℓ*-dimensional component is at most* $d^{2^{n-1-\ell}}$.

Proof. We proceed by induction on $s \overset{\text{def}}{=} n - \ell$.

Base case $s = 1$**.** Any $(n-1)$ dimensional component of $\mathtt{X}$ simply corresponds to an $\mathbb{F}_q$-factor of the polynomial $f_1(\bar{\mathbf{x}})$ and thereofore its degree is bounded by d, as required.

Induction step. Now any $(\ell - 1)$-dimensional component $\mathtt{X}_i$ of $\mathtt{X}$ is obtained by the intersection of a component $\widehat{\mathtt{X}}$ (obtained reviously during the computation) of dimension at most ℓ and a hypersurface $h(\bar{\mathbf{x}}) = 0$. The hypersurface $h(\bar{\mathbf{x}})$ is either one of the original input hypersurfaces $f_i(\bar{\mathbf{x}}) = 0$ or is of the form $\psi(\frac{\partial g}{\partial y_j}(\bar{\mathbf{y}}))$ for some birationally equivalent hypersurface $g(\bar{\mathbf{y}}) = 0$. In either case, the induction hypothesis implies that the degree d_h

of the hypersurface $h(\bar{\mathbf{x}}) = 0$ is bounded by $d_h \leq 2^{d^s}$. By induction hypothesis the degrees $\hat{d}$ of $\widehat{\mathtt{X}}$ is bounded by $\hat{d} \leq d^{2^s}$. By Bezout's theorem, the degree of $\mathtt{X}_i$ which is a component in their intersection is bounded by

$$\begin{aligned} d_h \cdot \hat{d} &\leq d^{2^s} \cdot d^{2^s} \\ &= d^{2^{s+1}}, \end{aligned}$$

as required.

□

Suppose that the corresponding birational hypersurface $g(\bar{\mathbf{y}}) = 0$ contains an absolutely irreducible $\mathbb{F}_q$-factor. Then the claim here is that $\mathtt{X}_i$ does indeed contain a rational point. If the dimension ℓ is zero, then the absolutely irreducible $\mathbb{F}_q$-factors of $g(y_1)$ are nothing but $\mathbb{F}_q$-points on $g(y_1) = 0$. The components output by the decomposition algorithm do not contain any singular varieties and thus no such point P is a singular point of $\mathtt{Y}_i$ and therefore $\phi(P)$ gives an $\mathbb{F}_q$-point on $\mathtt{X}_i$ as desired. If $\ell \geq 1$, then by theorem 6.3.5, $\mathtt{Y}_i$ contains a non-singular rational point P and therefore $\phi(P)$ gives a rational point on $\mathtt{X}_i$ as claimed.

Case II : $\mathtt{Y}_i$ has no absolutely irreducible $\mathbb{F}_q$-factors. We make use of the following lemma from the previous chapter.

Lemma 6.3.7. *Suppose that $h(\bar{\mathbf{y}}) \in \mathbb{F}_q[\bar{\mathbf{y}}]$ is $\mathbb{F}_q$-irreducible and it splits into absolutely irreducible factors $h_1(\bar{\mathbf{y}}), \ldots, h_t(\bar{\mathbf{y}})$ over some extension field $\mathbb{F}_{q^d}$ of $\mathbb{F}_q$. Then its absolutely irreducible factors $h_i(\bar{\mathbf{y}})$'s are all $\mathbb{F}_q$-conjugates of each other.*

Now consider a rational point P on the hypersurface $g(\bar{\mathbf{y}}) = 0$. Then P must be the zero of some $\mathbb{F}_q$-irreducible factor $h(\bar{\mathbf{y}})$ of $g(\bar{\mathbf{y}})$. That is $h(P) = 0$. Suppose $h(\bar{\mathbf{y}})$ splits completely over the extension field $\mathbb{K} \supsetneq \mathbb{F}_q$ into factors $h_1(\bar{\mathbf{y}}), \ldots, h_t(\bar{\mathbf{y}}) \in \mathbb{K}[\bar{\mathbf{y}}]$. Then P must be a rational point on some factor, say $h_1(\bar{\mathbf{y}})$, of $h(\bar{\mathbf{y}})$. Let $\sigma \in Gal_{\mathbb{K}/\mathbb{F}_q}$ be an automorphism of $\mathbb{K}$ mapping $h_1(\bar{\mathbf{y}})$ to $h_2(\bar{\mathbf{y}})$. Then since P is an $\mathbb{F}_q$-rational point, we have $\sigma(P) = P$. So P is also a zero of $h_2(\bar{\mathbf{y}})$ and hence P is a singular point on the surface $h(\bar{\mathbf{y}}) = 0$. Consequently P is also a singular point on the surface $g(\bar{\mathbf{y}}) = 0$.

Now a point P on $g(\bar{\mathbf{y}}) = 0$ is singular if and only if it is the common zero of the closed subset $\mathtt{Y}' \subsetneq \mathtt{Y}_i$ with defining equations

$$g(\bar{\mathbf{y}}) = \left(\frac{\partial g}{\partial y_1}\right)(\bar{\mathbf{y}}) = \ldots = \left(\frac{\partial g}{\partial y_{\ell+1}}\right)(\bar{\mathbf{y}}) = 0$$

By imposing the constraints $h_i(\bar{\mathbf{x}}) = \psi(\left(\frac{\partial g}{\partial y_1}\right)(\bar{\mathbf{y}}))$ on the algebraic set $\mathtt{X}_i$, our algorithm computes the preimage $\mathtt{X}' \subsetneq \mathtt{X}$ of $\mathtt{Y}'$. In this case then there is an $\mathbb{F}_q$-rational point P in $\mathtt{X}$ if and only if there is one in $\mathtt{X}'$, which our algorithm determines recursively, as required.

This completes the proof of correctness of our algorithm. We summarize it as a theorem.

Theorem 6.3.8. *Algorithm 1 is a deterministic algorithm which decides Solvability on an input consisting of a finite field $\mathbb{F}_q$ and polynomials $f_1, f_2, \cdots, f_m \in \mathbb{F}_q[x_1, x_2, \cdots, x_n]$ of total degree bounded by d in time poly($d^{c_n} \cdot m \cdot \log q$), where c_n is a constant that depends on n alone and is of size $n^{O(n)}$.*

Chapter 7

A blackbox derandomization of Primality Testing

Summary: We present a deterministic polynomial-time algorithm that determines whether an input number n is prime or composite.

7.1 Introduction

The primality testing problem is to determine whether an input number n is prime or composite. It is one of the fundamental problems in algorithmic number theory with important applications in cryptography and elsewhere. An unconditional deterministic polynomial time algorithm for primality testing was first presented in [AKS04]. In this chapter, we present a variant of the AKS algorithm for testing primality. The basis of the AKS algorithm [AKS04] is an identity which prime numbers and only prime numbers satisfy.

Identity. *An integer $n \geq 2$ is prime if and only if*

$$(x-1)^n = (x^n - 1) \pmod{n}$$

The variant of the AKS algorithm [AKS04] that we present in this chapter is motivated by the connection between derandomization of identity testing and proving arithmetic circuit lower bounds. Agrawal [Agr05], following the work of Kabanets and Impagliazzo [IK03], observed that *a black-box derandomization* of the identity testing problem is equivalent to proving arithmetic circuit lower bounds.

7.1.1 Black-box derandomization in general.

To understand the notion of black-box derandomization, recall that a randomized algorithm $\mathcal{A}$ for a language $\mathrm{L} \subseteq \{0,1\}^*$ takes as input a pair of strings $\langle x, r\rangle$ and accepts with *high probability* (over the random choice of r) if and only if $x \in \mathrm{L}$. It is known that for any $n \geq 1$, there exists a set S of size $\mathrm{poly}(n)$ of strings $\{r_1, r_2, \ldots, r_t\}$ such that for any $x \in \{0,1\}^*$ of length n,

$$x \in L \quad \text{iff} \quad \text{Majority}(\mathcal{A}(x, r_1), \mathcal{A}(x, r_2), \ldots, \mathcal{A}(x, r_t)) = 1.$$

A *black-box derandomization of the algorithm* $\mathcal{A}$ is an explicit (deterministic polynomial time) computation of such a set S.

7.1.2 Black-box derandomization of identity testing.

The underlying randomized algorithm for identity testing is the well-known Schwarz-Zippel algorithm which evaluates the given polynomial at a randomly chosen point P. Then the black-box derandomization problem for identity testing is the following problem: for a fixed field $\mathbb{F}$, given an integer $s \geq 1$ construct a set A of points of size $poly(s)$ in deterministic polynomial time so that any circuit of size s over $\mathbb{F}$ computes the zero polynomial if and only if it evaluates to zero at all points in the set A (the coordinates of each of the points in A are allowed to lie in some small ($poly(s)$-dimensional) ring extension R of $\mathbb{F}$). In this chapter, we give an explicit construction of the set A of evaluation points which derandomizes the identity for primality testing.

7.2 A randomized algorithm for primality

A randomized algorithm for primality was proposed by Agrawal and Biswas [AB03] and it was based on the following identity which prime numbers and only prime numbers satisfy. [1] For any natural number n let

$$P_n(x) \stackrel{\text{def}}{=} (x-1)^n - (x^n - 1)$$

Identity. *An integer $n \geq 2$ is prime if and only if*

$$P_n(x) = 0 \pmod{n} \tag{7.1}$$

[1] The algorithm we present here is a minor variation of the algorithm of Agrawal and Biswas [AB03].

Proof. For $0 < i < n$, the coefficient of x^i in $P_n(x) = ((x-1)^n - (x^n - 1))$ is $(-1)^{n-i}\binom{n}{i}$. Now if n is prime, $\binom{n}{i} \equiv 0(\mathrm{mod}\ n)$ and hence all the coefficients are zero.
If n is composite: consider a prime q that is a factor of n and let $q^k || n$. Then q^k does not divide $\binom{n}{q}$ and hence the coefficient of x^q is not zero (mod n). Thus $((x-1)^n - (x^n - 1))$ is not identically zero over $\mathbb{Z}/n\mathbb{Z}$. □

Thus given an n as input, one could compute whether the congruence (7.1) is satisfied or not. However, this takes time $\Omega(n)$ because we need to evaluate n coefficients in the LHS in the worst case. Therefore, to make it feasible we evaluate (7.1) modulo a randomly chosen polynomial $Q(x)$ of degree $r = (\log n)^2$. Our algorithm then consists of verifying congruences of the form

$$P_n(x) = 0 \pmod{Q(x), n}. \tag{7.2}$$

From the Identity 7.2, it is immediate that all primes n satisfy the above congruence for all choices of $Q(x)$; however some composites n may also satisfy congruence (7.2) for a few choices of $Q(x)$. The above congruence takes $O(r^2 \log^3 n)$ time for verification (lhs is evaluated by repeated squaring), or even better $O(r \log^2 n)$ if Fast Fourier Multiplication [Knu81] is used.

```
Algorithm:  Randomized Primality test
Input:  integer $n > 1$

  1. pick a random monic $Q(x) \neq 0 \in (\mathbb{Z}/n\mathbb{Z})[x]$ of degree $r = 16(\log n)^5$.

  2. if ( $(x-1)^n \not\equiv (x^n - 1))$ (mod $x^{15} \cdot Q(x), n)$) output COMPOSITE;

  3. output PRIME;
```

The running time of the algorithm is clearly *poly*$(\log n)$. The next theorem bounds the error probability of the algorithm.

Theorem 7.2.1. *If the input n to the above algorithm is prime then our algorithm outputs PRIME. If n is composite then the algorithm outputs PRIME with probability at most $\frac{2}{3}$.*

Proof. If n is prime then by identity 7.2, it satisfies

$$(x-1)^n \not\equiv (x^n - 1)) \pmod{Q(x), n})$$

for *all* polynomials $Q(x)$ and hence the algorithm outputs PRIME. Now assume that n is composite. Let

$$P_n(x) \overset{\text{def}}{=} (x-1)^n - (x^n - 1)$$

We first observe that n cannot have any prime factors smaller than 15.

Claim 7.2.1.1. *If n the algorithm outputs PRIME then n cannot have any proper prime factors less than* 15.

Proof of Claim 7.2.1.1. Suppose if possible a prime $q \leq 15$ divides n. Since the algorithm returns PRIME, we must have:

$$(x-1)^n \equiv (x^n - 1) \pmod{x^{15}, n}.$$

This happens if and only if

$$(-1)^i \binom{n}{i} \equiv 0 \pmod{n} \; \forall 1 \leq i < 15.$$

Setting $i := q < n$, we get

$$\binom{n}{q} \equiv 0 \pmod{n}$$

As in the proof of Identity 7.2 this cannot happen for if q^a is the largest power of q which divides n, then $\binom{n}{q}$ is divisible by q^{a-1} but *not* by q^a. Therefore $\binom{n}{q} \not\equiv 0 \pmod{q^a}$ which implies $\binom{n}{q} \not\equiv 0 \pmod{n}$, a contradiction. □

Let p be a prime divisor of n such that p^a exactly divides n. Then by the proof of identity 7.2, there exists a polynomial $A(x) \neq 0 \in \mathbb{F}_p[x]$ of degree less than n such that

$$P_n(x) \equiv p^\ell \cdot A(x) \pmod{p^a}$$

for some $\ell < a$. Thus we have

$$P_n(x) \equiv 0 \pmod{Q(x), n} \quad \Rightarrow \quad A(x) \equiv 0 \pmod{Q(x), p}$$

Now the analysis of [AB03] applies so that the probability that the algorithm outputs COMPOSITE is at least $\frac{2}{3}$.

□

7.3 Derandomization of Primality Testing Algorithm.

We give a derandomization of the randomized algorithm for primality mentioned in the previous section. Ours is a black box derandomization of the randomized test presented above in the sense that we specify a set R of polynomials,

$$R = \{(x+a)^r - b \ : \ 1 \leq r \leq 16(\log n)^5, \ 0 \leq a \leq 8(\log n)^{7/2}, \ 0 \leq b \leq 1\}$$

so that by letting $Q(x)$ run over the set of polynomials in R, for composite n, the congruence 7.2 fails to hold at least once. The deterministic primality test then is very simple and is given below. Here and henceforth, we shall denote by $[n]$ the set of first $(n+1)$ non-negative integers $\{0, 1, \ldots, n\}$. [2]

```
Algorithm:  Deterministic Primality test
Input:  integer $n > 1$

  1. let $R = \{(x+a)^r - b \ : r \in [16(\log n)^5], a \in [8(\log n)^{7/2}], \ b \in \{0,1\}\}$.

  2. for all $Q(x) \in R$ do
       if ( $(x-1)^n \not\equiv (x^n - 1)$ (mod $x^{15} \cdot Q(x), n$) ) output COMPOSITE;

  3. output PRIME;
```

Clearly, this is a deterministic polynomial time algorithm. In the remainder of this chapter, we establish its correctness through a sequence of lemmas. That is, we will establish the following theorem -

Theorem 7.3.1. *The algorithm above returns PRIME if and only if n is prime.*

Lemma 7.3.2. *If n is prime, the algorithm returns PRIME.*

Proof: When the input n is a prime, by Identity 7.2, we have that

$$(x-1)^n \equiv (x^n - 1) \ (\text{mod } Q(x), n)$$

[2] The algorithm and its proof are based on a talk given by Manindra Agrawal at the American Institute of Mathematics. The transcript is available at `http://www.aimath.org/WWN/primesinp/articles/html/41a/` .

holds for *all* $Q(x)$ and hence our algorithm outputs PRIME as expected. ∎

The converse of the above lemma requires somewhat more work. For the purpose of subsequent analysis assume that n is a composite. We first observe that our algorithm works correctly for very small n (≤ 20).

Lemma 7.3.3. *For $n \leq 20$, the algorithm above returns PRIME if and only if n is prime.*

Proof: If $n \leq 20$ then it holds that $n \leq 16(\log n)^5$. For n that small, the degree of the polynomial $((x-1)^n - (x^n - 1))$ is smaller than that of the largest quotienting polynomial $Q(x)$ and hence for such a $Q(x)$,

$$(x-1)^n \equiv (x^n - 1) \pmod{Q(x), n}$$

implies that

$$(x-1)^n \equiv (x^n - 1) \pmod{n}.$$

By Identity 7.2 this happens if and only if n is prime. ∎

Having shown this, for the rest of the analysis we can assume that $n > 16(\log n)^5$. Next we observe that n cannot have any *small* ($\ll (\log n)^5$) prime factors.

Lemma 7.3.4. *If the algorithm outputs PRIME, then n does not have any prime factors smaller than* $16(\log n)^5$.

Proof: Suppose if possible $q \leq 16(\log n)^5$ divides n. Let $r = 16(\log n)^5$. Since the algorithm returns PRIME, (picking $a = b = 0$) we must have:

$$(x-1)^n \equiv (x^n - 1) \pmod{x^r, n}.$$

This happens if and only if

$$(-1)^i \binom{n}{i} \equiv 0 \pmod{n}\ \forall 1 \leq i < r.$$

Setting $i := q$, we get

$$\binom{n}{q} \equiv 0 \pmod{n}$$

As in the proof of Identity 7.2 this cannot happen for if q^a is the largest power of q which divides n, then $\binom{n}{q}$ is divisible by q^{a-1} but *not* by q^a. Therefore $\binom{n}{q} \not\equiv 0 \pmod{q^a}$ which implies $\binom{n}{q} \not\equiv 0 \pmod{n}$, a contradiction. ∎

Next we show that n must be square-free.

Lemma 7.3.5. *If the algorithm outputs PRIME, then n must be square-free.* [3]

Proof: Suppose if possible that for some prime p, $p^2|n$. Since the algorithm outputs PRIME, we have

$$(x-1)^n \equiv (x^n - 1) \pmod{x^r - 1, n} \ \ \forall 1 \leq r \leq 16(\log n)^5$$

Substituting x^k for x in

$$(x-1)^n \equiv (x^n - 1) \pmod{x^r - 1, n},$$

we get

$$(x^k - 1)^n \equiv (x^{kn} - 1) \pmod{x^{kr} - 1, n}.$$

But $(x^r - 1)|(x^{kr} - 1)$ and so we have

$$(x^k - 1)^n \equiv (x^{kn} - 1) \pmod{x^r - 1, n}.$$

Multiplying we get

$$\prod_{k=1}^{r-1}(x^k - 1)^n \equiv \prod_{k=1}^{r-1}(x^{kn} - 1) \pmod{x^r - 1, n},$$

so that

$$r^n \equiv r (\text{mod } n) \ \text{ for all } r \leq 16(\log n)^5 \tag{7.3}$$

Then by Lemma 7.3.4 we have $p > 16(\log n)^5$. So by equation 7.3, for all $r \leq 16(\log n)^5$ we have:

$$r^{n-1} \equiv 1 \pmod{p^2}$$

Also, since the group $(\mathbb{Z}/p^2\mathbb{Z})^*$ has order $p(p-1)$ we have

$$r^{p(p-1)} \equiv 1 \pmod{p^2}$$

And so

$$r^{\gcd(n-1, p(p-1))} \equiv 1 \pmod{p^2}$$

$$\Rightarrow r^{p-1} \equiv 1 \pmod{p^2}$$

[3] *This lemma is based on a proof by Hendrik Lenstra Jr. showing that a number which passes Fermat's test for primality: $a^n = a \pmod n$, for a few small values of a, has to be square-free.*

Thus all the r's between 1 and $16(\log n)^5$ are roots of the polynomial $(x^{p-1}-1) \in \mathbb{Z}/p^2\mathbb{Z}[x]$. Now the polynomial $x^{p-1}-1$ viewed over the field $\mathbb{Z}/p\mathbb{Z}$ is squarefree and so by the well-known Hensel lifting lemma, every root of $(x^{p-1}-1)$ in $\mathbb{Z}/p\mathbb{Z}$ lifts to a unique root of $(x^{p-1}-1)$ in $\mathbb{Z}/p^2\mathbb{Z}$. Consequently it has exactly $(p-1)$ distinct roots in $\mathbb{Z}/p^2\mathbb{Z}$.

Also observe that if r_1 and r_2 satisfy $x^{p-1} \equiv 1 \pmod{p^2}$ then so does their product $r_1 \cdot r_2$. This means that all the $(16 \log^5 p)$-smooth numbers are solutions of

$$x^{p-1} \equiv 1 \pmod{p^2}.$$

However, by the bound in [CEG83], there are more than p numbers smaller than p^2 which are $(16 \log^5 p)$-smooth. This gives us a contradiction. n must therefore be squarefree.

▮

We will be needing the following simple fact about the lcm of first m numbers (cf. [Nai82] for a proof).

Lemma 7.3.6. *Let* $\textsc{Lcm}(m)$ *denote the lcm of first* m *numbers. For* $m \geq 8$*:*

$$\textsc{Lcm}(m) \geq 2^m.$$

We next show the existence of a small r such that the order of n modulo r, $\mathrm{o}_r(n)$ is large.

Lemma 7.3.7. *There exist an* $r \leq 16 \log^5 n$ *such that* $\gcd(n,r) = 1$ *and* $\mathrm{o}_r(n) > 4 \log^2 n$*.*

Proof. Let r_1, r_2, …, r_t be all numbers coprime to n such that $\mathrm{o}_{r_i}(n) \leq 4 \log^2 n$. Each of these numbers must divide the product

$$\prod_{i=1}^{4\log^2 n} (n^i - 1) < n^{16 \log^4 n} = 2^{16 \log^5 n}.$$

By Lemma 7.3.6, the lcm of first $16 \log^5 n$ is at least $2^{16 \log^5 n}$ and moreover by lemma 7.3.4 they are all coprime to n. Therefore there must exist a number $r \leq 16 \log^5 n$ such that $\gcd(r,n) = 1$ and $\mathrm{o}_r(n) > 4 \log^2 n$. □

Since $\mathrm{o}_r(n) > 1$, there must exist a prime divisor p of n such that $\mathrm{o}_r(p) > 1$. Since $(n,r) = 1$ (by lemma 7.3.4), $p, n \in Z_r^*$. Numbers p and r will be fixed in the remainder of this section. Also, let $\ell = 8(\log n)^{7/2}$.

For $Q(x)$ of the form $(x+a)^r-1$, the algorithm verifies ℓ equations. Since the algorithm does not output COMPOSITE in this step, we have:

$$(x-1)^n = x^n - 1 \ (\text{mod } (x+a)^r - 1, n)$$

Substituting $(x-a)$ for x throughout we get

$$(x-a-1)^n = (x-a)^n - 1 \ (\text{mod } x^r - 1, n)$$

for every a, $1 \le a \le \ell$. Via induction on a, this implies that

$$(x-a)^n = (x)^n - a \ (\text{mod } x^r - 1, n)$$

for every a, $1 \le a \le \ell$. This implies:

$$(x-a)^n = x^n - a \ (\text{mod } x^r - 1, p) \tag{7.4}$$

for $1 \le a \le \ell$. By the Identity 7.2, we have:

$$(x-a)^p = x^p - a \ (\text{mod } x^r - 1, p) \tag{7.5}$$

for $1 \le a \le \ell$. Thus n behaves like prime p in the above equation. We give a name to this property:

Definition 7.3.8. For polynomial $f(x)$ and number $m \in \mathcal{N}$, we say that m is *introspective* for $f(x)$ if

$$[f(x)]^m = f(x^m) \ (\text{mod } x^r - 1, p).$$

It is clear from equations (7.4) and (7.5) that both n and p are introspective for $(x-a)$ for $1 \le a \le \ell$.

The following lemma shows that introspective numbers are closed under multiplication:

Lemma 7.3.9. *If m and m' are introspective numbers for $f(x)$ then so is $m \cdot m'$.*

Proof. Since m is introspective for $f(x)$ we have:

$$[f(x)]^{m \cdot m'} = [f(x^m)]^{m'} \ (\text{mod } x^r - 1, p).$$

Also, since m' is introspective for $f(x)$, we have (after replacing x by x^m in the equation):

$$\begin{aligned} [f(x^m)]^{m'} &= f(x^{m \cdot m'}) \ (\text{mod } x^{m \cdot r} - 1, p) \\ &= f(x^{m \cdot m'}) \ (\text{mod } x^r - 1, p) \ (\text{since } x^r - 1 \text{ divides } x^{m \cdot r} - 1). \end{aligned}$$

Putting togather the above two equations we get:

$$[f(x)]^{m\cdot m'} = f(x^{m\cdot m'}) \pmod{x^r - 1, p}.$$

□

For number m, the set of polynomials for which it is introspective is also closed under multiplication:

Lemma 7.3.10. *If m is introspective for $f(x)$ and $g(x)$ then it is also introspective for $f(x)\cdot g(x)$.*

Proof. We have:

$$\begin{aligned} [f(x)\cdot g(x)]^m &= [f(x)]^m \cdot [g(x)]^m \\ &= f(x^m)\cdot g(x^m) \pmod{x^r - 1, p}. \end{aligned}$$

□

The above two lemmas togather imply that every number in the set $I = \{n^i \cdot p^j \mid i, j \geq 0\}$ is introspective for every polynomials in the set $P = \{\prod_{a=1}^{\ell}(x-a)^{e_a} \mid e_a \geq 0\}$. We now define two groups based on these sets that will play a crucial role in the proof.

The first group is the set of all residues of numbers in I modulo r. This is a subgroup of Z_r^* since, as already observed, $(n, r) = (p, r) = 1$. Let G be this group and $|G| = t$. G is generated by n and p modulo r and since $\mathrm{o}_r(n) > 4\log^2 n$, $t > 4\log^2 n$.

The second group is the set of all residues of polynomials in P modulo $h(x)$ and p where $h(x)$ is an irreducible factor of $\frac{x^r-1}{x-1}$ over $\mathbb{F}_p$ of degree > 1 (such an h will always exist since $\mathrm{o}_r(p) > 1$ by our choice of p). Let $\mathcal{G}$ be this group. This group is generated by $x-1$, $x-2$, …, $x-\ell$ in the field $\mathbb{F} = \mathbb{F}_p[x]/(h(x))$ and is a subgroup of the multiplicative group of $\mathbb{F}_p[x]/(h(x))$.

If n is not a power of p then the size of $\mathcal{G}$ is effectively determined by the size of G as shown by the lemmas below.

Lemma 7.3.11. $|\mathcal{G}| \geq \min\{2^t - 1, 2^\ell\}$.

Proof. We show that any two distinct polynomials of degree less than t in P will map to different elements in $\mathcal{G}$. Let $f(x)$ and $g(x)$ be two such polynomials in P. Suppose

$f(x) = g(x)$ in the field $\mathbb{F}$. Let $m \in I$. We also have $[f(x)]^m = [g(x)]^m$ in $\mathbb{F}$. Since m is introspective for both f and g, we get:

$$f(x^m) = g(x^m)$$

in $\mathbb{F}$. This implies that x^m is a root of the polynomial $Q(y) = f(y) - g(y)$ for every $m \in G$. Since the size of G is t there will be t distinct such roots of $Q(y)$ in $\mathbb{F}$. However, the degree of polynomial $Q(y)$ is less than t by the choice of f and g. This is a contradiction and therefore, $f(x) \neq g(x)$ in $\mathbb{F}$.

If $t \leq \ell$ then there exist at least $2^t - 1$ polynomials of degree less than t in P (all subsets of first t $(x-a)$'s except the one containing all of them). And if $t > \ell$ then there exist at least 2^ℓ such polynomials (all subsets of ℓ $(x-a)$'s). □

Lemma 7.3.12. *If n is not a power of p then $|\mathcal{G}| < n^{2\sqrt{t}}$.* [4]

Proof. Consider the following subset of I:

$$\hat{I} = \{n^i \cdot p^j \mid 0 \leq i, j \leq \sqrt{t}\}.$$

If n is not a power of p then the set $\hat{I}$ has more than t distinct numbers. Since $|G| = t$, at least two numbers in $\hat{I}$ must be equal modulo r. Let these be m_1 and m_2 with $m_1 > m_2$. So we have:

$$x^{m_1} = x^{m_2} \pmod{x^r - 1}.$$

Let $f(x) \in P$. Then,

$$\begin{aligned} [f(x)]^{m_1} &= f(x^{m_1}) \ (\text{mod } x^r - 1, p) \\ &= f(x^{m_2}) \ (\text{mod } x^r - 1, p) \\ &= [f(x)]^{m_2} \ (\text{mod } x^r - 1, p). \end{aligned}$$

This implies

$$[f(x)]^{m_1} = [f(x)]^{m_2}$$

in the field $\mathbb{F}$. Therefore, $f(x) \in \mathcal{G}$ is a root of the polynomial $Q'(Y) = Y^{m_1} - Y^{m_2}$ in the field $\mathbb{F}$.As $f(x)$ is an arbitrary and non-zero element of $\mathcal{G}$, we have that the polynomial $Q'(Y)$ has at least $|\mathcal{G}|$ distinct roots in F. The degree of $Q'(Y)$ is $m_1 \leq (np)^{\sqrt{t}} < n^{2\sqrt{t}}$ (since $n > p$). This shows $|\mathcal{G}| < n^{2\sqrt{t}}$. □

[4] *This version of the proof is due to Adam Kalai, Amit Sahai and Madhu Sudan. It makes the proof for both the upper and lower bounds for $\mathcal{G}$ similar.*

Armed with the estimates on the size of $\mathcal{G}$, we are now ready to prove the correctness of the algorithm:

Lemma 7.3.13. *If the algorithm returns PRIME then n is prime.*

Proof. Suppose that the algorithm returns PRIME. Lemma 7.3.11 implies that for $t = |G|$ and $\ell = 2\sqrt{\phi(r)}\log n$:

$$\begin{aligned}
|\mathcal{G}| &\geq \min\{2^t - 1, 2^\ell\} \\
&= \min\{2^t - 1, n^{8(\log n)^{5/2}}\} \\
&\geq \min\{2^t - 1, n^{2\sqrt{t}}\} \text{ (since } t < r \leq 16(\log n)^5 \text{)} \\
&\geq \min\{2^{2\sqrt{t}\log n}, n^{2\sqrt{t}}\} \text{ (since } t > 4\log^2 n) \\
&\geq n^{2\sqrt{t}}.
\end{aligned}$$

By Lemma 7.3.12, $|\mathcal{G}| < n^{2\sqrt{t}}$ if n is not a power of p. Therefore, $n = p^k$ for some $k > 0$. By lemma 7.3.5, n is squarefree. Therefore, $k \leq 1$ and $n = p$. □

This completes the proof of theorem.

7.4 Summary

In this chapter, we presented a variant of the deterministic primality test of [AKS04]. The running time of the algorithm even though polynomially bounded is larger than the running time of [AKS04]. However the algorithm presented is a black-box derandomization of its corresponding randomized algorithm.

Chapter 8

Conjectures and Open Problems

> *“ As long as a branch of science offers an abundance of problems, so long is it alive; a lack of problems foreshadows extinction or the cessation of independent development. Just as any human undertaking pursues certain objects, so also mathematical research requires its problems. It is by the solution of problems that the investigator tests the temper of his steel; he finds new methods and new outlooks, and gains a wider and freer horizon.”*
>
> - David Hilbert, *Mathematical Problems*, ICM Paris, 1900.

8.1 Introduction

For the benefit of the interested researcher looking for some elegant (and approachable) problems to work on, we note down some open problems in this chapter. The reader is also referred to the open problems paper by Adleman and McCurley [AM94] for an excellent collection of elegant, well-studied open problems in algorithmic number theory. We avoid open-ended problems and in doing so, present some algorithmic problems as conjectures, where appropriate. For the sake of clarity we chose to state a specific version of a problem rather than a general one. For example questions about curves have natural generalizations to arbitary dimensional varieties, questions for algebras have natural generalizations to arbitary rings, complexity-theoretic questions over finite fields have natural recursion-theoretic analogs over the field of rationals $\mathbb{Q}$ and conversely, questions and conjectures over rationals have natural analogs for the rational function field $\mathbb{F}_p(y)$ over $\mathbb{F}_p$ and so on.

8.2 Identity testing

Recall from chapter 1 that the identity testing problem is the following: given a field $\mathbb{F}$ and an arithmetic circuit $\mathcal{C}$ over $\mathbb{F}$, determine if the polynomial computed by it is the identically zero polynomial. Identity Testing is arguably the most important derandomization problem. Several interesting special cases remain unresolved. These special cases are obtained by placing appropriate restrictions on the input circuit $\mathcal{C}$.

Case-1. **Multilinear Formula Identity Testing:** An arithmetic circuit $\mathcal{C}$ is said to be a *formula* if the fanout of every gate is one. An arithmetic formula is said to be multilinear if the polynomial computed by each gate of the formula is multilinear.

Open Problem. ([Raz04]): Devise a deterministic sub-exponential time algorithm for identity testing of multilinear formulas.

Case-2. **Identity Testing for Bounded Depth Circuits:**

Open Problem. For a fixed $d \geq 3$, devise a deterministic polynomial-time algorithm for identity testing of arithmetic circuits of depth d.

In this connection we reproduce the following conjecture by Agrawal [Agr05] which if true would derandomize bounded depth identity testing and would moreover imply the strongest lower bound known yet.

Conjecture. [Agr05]: Let $\mathbb{F}$ be a field and $\mathcal{C}(x_1, \ldots, x_n)$ be an arithmetic circuit of size s and depth d computing a polynomial in n variables over $\mathbb{F}$. Let $r \geq s^{4d}$ be a prime. Then $\mathcal{C}$ computes the identically zero polynomial over $\mathbb{F}$ if and only if

$$\mathcal{C}(y^{k^0}, y^{k^1}, \ldots, y^{k^{n-1}}) = 0 \pmod{y^r - 1} \quad \forall k \in [r]$$

Remark. Agrawal [Agr05] shows that if the above conjecture is true it would imply the following arithmetic circuit lower bound: for every $\epsilon > 0$ there exists a multilinear polynomial computable in E that cannot be computed by a nonuniform family of arithmetic circuits with unbounded fanin addition gates of size $s^{d-\epsilon}$ and depth $(d - \epsilon) \log s$.

In this connection, we mention that the best arithmetic circuit lower bounds are that of Karpinski and Grigoriev [GK98] which shows that *for a fixed finite field* $\mathbb{F}_p$, computing the determinant of an $n \times n$ matrix requires exponential size $\mathbb{F}_p$-circuits. For algebraically closed fields, we do not have lower bounds even for depth-3 circuits and in particular the following problem is open.

Conjecture. Computing the determinant of an $n \times n$ matrix requires exponential size $\mathbb{C}$-circuits of depth three.

There is also a very interesting conjecture due to Dvir and Shpilka [DS05] regarding the structure of $\Sigma\Pi\Sigma$ identities.

Conjecture. (Dvir and Shpilka, [DS05]) Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma$ circuit *over the field* $\mathbb{C}$ *of complex numbers.* We will call $\mathcal{C}$ to be *minimal* if no proper subset of the multiplication gates of $\mathcal{C}$ sums to zero. We say that $\mathcal{C}$ is *simple* if there is no linear function that appears in all the multiplication gates (up to a multiplicative constant). *Rank* of $\mathcal{C}$ is the rank of the linear forms appearing in $\mathcal{C}$. If $\mathcal{C}$ computes the the identically zero polynomial then the rank of $\mathcal{C}$ can at most be linear in the fanin k of the topmost addition gate.

8.3 Computing rational points on curves and varieties over a finite field.

Open Problem: Given a finite field $\mathbb{F}_q$ and a bivariate polynomial $f(x, y) \in \mathbb{F}_q[x, y]$ compute an $\mathbb{F}_q$-rational point, if it exists, on the curve $f(x, y) = 0$ in deterministic polynomial time.

Remark.

i. This problem has a natural generalization to higher dimensional varieties over finite fields.

ii. This problem admits a random polynomial-time algorithm (cf. [dW06]).

iii. Adleman and McCurley [AM94] posed this problem for the particular case of elliptic curves. Christiaan Van Der Woestijne [dW06] recently settled this for elliptic curves.

8.4 Quantified Formulae in bounded number of variables over $\mathbb{F}_q$

Understanding the structure of the solution space of a system of polynomial equations over a finite field is a fundamental number-theoretic problem. In particular consider the following general problem: Let $\mathbb{F}_q$ be a finite field and $f_1, f_2, \cdots, f_m \in \mathbb{F}_q[x_1, x_2, \cdots, x_n]$ be a collection of m polynomials where for each polynomial the total degree is at most d. Let $Q_1, Q_2, \cdots, Q_n$ be some sequence of quantifiers, i.e. each Q_i is either "$\exists$" or "$\forall$". Determine the truth value of the following statement:

$$Q_1 x_1 Q_2 x_2 \cdots Q_n x_n \text{ such that } f_1(x_1, \cdots, x_n) = \cdots = f_m(x_1, \cdots, x_n) = 0?$$

It is easy to see that this problem is PSPACE-complete. We pose the following problem: What is the computational complexity of this problem when the number n of variables (and hence also the number of quantifiers) is bounded? In this dissertation, we saw that if all the quantifiers are existential than the problem is in P . For concreteness, we state the problem for $n = 2$ variables.

Open Problem: Given a finite field and a polynomial $f(x, y) \in \mathbb{F}_q[x, y]$ determine if the following statement is true:

$$\forall (y \in \mathbb{F}_q) \quad \exists (x \in \mathbb{F}_q) \quad f(x, y) = 0.$$

Remark.

i. The Chebotarev density theorem over finite fields implies that this problem admits an algorithm with running time poly$(d^d \log q)$, where d is the degree of $f(x, y)$.

ii. We do not know even a randomized polynomial-time algorithm for this problem.

iii. This problem has a natural analog over the field $\mathbb{Q}$ of rational numbers. We make the following conjecture which if true would immediately imply an efficient algorithm over rationals.

    Conjecture: For a polynomial $f(x, y) \in \mathbb{Z}[x, y]$ which is monic in x the statement

    $$\forall (y \in \mathbb{Q}) \quad \exists (x \in \mathbb{Q}) \quad f(x, y) = 0$$

    is true if and only if $f(x, y)$ has a factor which is linear in x.

8.5 $\mathbb{F}$-algebra isomorphism.

Open Problem. Given two algebras R_1 and R_2 of dimension d over a finite field $\mathbb{F}_p$, determine if the two algebras are isomorphic.

Remark.

i. Saxena [Sax06] shows that Graph Isomorphism reduces to $\mathbb{F}$-algebra isomorphism which in turn reduces to Equivalence of Cubic Forms.

ii. When the two algebras constitute finite fields, Hendrik Lenstra [Len91] gives a *deterministic* polynomial-time algorithm that explicitly computes the isomorphism $\phi : R_1 \mapsto R_2$.

iii. We do not know of an efficient algorithm even when the algebras are represented verbosely by their addition and multiplication tables. That is, we do not know of an algorithm with running time poly(p^d).

iv. This problem has a natural analogue over the field $\mathbb{Q}$ of rational numbers. We make the following conjecture which if true would immediately imply a decidable algorithm over rationals.

   Conjecture. Local-Global principle for $\mathbb{Q}$-algebra isomorphism: Let R_1 and R_2 be two algebras of dimension d over $\mathbb{Q}$, the field of rational numbers. Then R_1 is isomorphic to R_2 if and only if $R_1 \equiv R_2$ over the field of real numbers and $R_1 \equiv R_2$ over the field $\mathbb{Q}_p$ of p-adic numbers for every prime p.

Appendix A

Reduction of GI to Ring Isomorphism

In this appendix we reproduce the construction of Saxena [Sax06] which reduces Graph Isomorphism to isomorphism testing of local $\mathbb{F}_p$-algebras. The construction works for any $p \geq 5$. For concreteness we set $p = 5$.

Let G be an undirected graph with n vertices and no self loops. The construction gives a local commutative $\mathbb{F}_5$-algebra. They associate variables to each vertex (x-variable) and capture the "connectivity" of the graph by defining the edges-polynomial – $\sum_{(u,v) \text{ is an edge}} x_u x_v$ – as zero in the ring.

Define the following commutative $\mathbb{F}_5$-algebra:

$$R(G) := \mathbb{F}_5[x_1, \ldots, x_n]/\mathcal{I}$$

where, ideal $\mathcal{I}$ has the following relations:

Case-1. x's are nilpotents of degree 2, i.e., for all $i \in [n]$: $x_i^2 = 0$.

Case-2. the edges-polynomial is zero, i.e., $\sum_{\substack{1 \leq i < j \leq n \\ (i,j) \in E(G)}} x_i x_j = 0$.

Case-3. all cubic terms are zero, i.e., for all $i, j, k \in [n]$: $x_i x_j x_k = 0$.

Suppose (i_0, j_0) is an edge in G such that $1 \leq i_0 < j_0 \leq n$. Then the additive structure of the ring is:

$$(R(G), +) = \mathbb{F}_5 \cdot 1 \oplus \bigoplus_{i \in [n]} \mathbb{F}_5 \cdot x_i \oplus \bigoplus_{\substack{i < j \in [n] \\ (i,j) \neq (i_0, j_0)}} \mathbb{F}_5 \cdot (x_i x_j)$$

Thus, the dimension of the ring over $\mathbb{F}_5$ is $\binom{n+1}{2}$. Multiplication satisfies the associative law simply because the product of any three *variables* (in any order) is zero. Also, $R(G)$ is a local commutative $\mathbb{F}_5$-algebra.

Observe that if $G_1 \cong G_2$ then any graph isomorphism ϕ induces a natural isomorphism between rings $R(G_1)$ and $R(G_2)$. So we only have to prove the converse:

Lemma A.0.1. *Let G and G' be two undirected graphs having no self-loops. Further, assume that graphs G and G' are not a disjoint union of a clique and a set of isolated vertices. Then, $R(G) \cong R(G')$ implies $G \cong G'$.*

Proof. Suppose ϕ is an isomorphism from $R(G) \to R(G')$. Let

$$\phi(x_i) = c_{i,0} + c_{i,1}x_1 + \ldots + c_{i,n}x_n + (\text{quadratic terms}). \tag{A.1}$$

where all $c_{i,j}$'s in the coefficients are in $\mathbb{F}$.

By squaring the above we get:

$$0 = \phi(x_i^2) = \phi(x_i)^2 = c_{i,0}^2 + (\text{linear and quadratic terms})$$

which means that $c_{i,0} = 0$. The next observation about ϕ is that there is at most one nonzero linear term in $\phi(x_i)$. Let $C_i = \{j \in [n] \mid c_{i,j} \neq 0\}$ be of size > 1. Then $\phi(x_i)^2 = 0$ gives:

$$\sum_{j<k\in C_i} (2c_{i,j}c_{i,k})x_jx_k \;=\; 0 \text{ in } R(G')$$

We know that in $R(G')$ the quadratic relations are $x_i^2 = 0$ and $\sum_{\substack{1\leq i<j\leq n\\(i,j)\in E(G')}} x_ix_j = 0$. This means that the above equation holds only if there is a $\lambda \in \mathbb{F}$:

$$\sum_{\substack{1\leq j<k\leq n\\ j,k\in C_i}} (2c_{i,j}c_{i,k})x_jx_k = \lambda \cdot \sum_{\substack{1\leq i<j\leq n\\(i,j)\in E(G')}} x_ix_j = 0$$

This equality interpreted in graph terms means that G' is a union of a clique on C_i and a set of $(n - \#C_i)$ isolated vertices (remember that $2 \neq 0$ in $\mathbb{F}$). This we ruled out in the hypothesis, thus size of $C_i \leq 1$. If $\#C_i = 0$ then for any j, $\phi(x_ix_j) = 0$ which contradicts the assumption that ϕ is an isomorphism. Thus, for all $i \in [n]$, $\#C_i = 1$. Define a map $\pi : [n] \to [n]$ such that the nonzero linear term occurring in $\phi(x_i)$ is $x_{\pi(i)}$.

Suppose π is not a permutation on $[n]$ then there are $i \neq j$ such that $\pi(i) = \pi(j)$. But then there will exist $a, b \in \mathbb{F}^*$ such that there is no nonzero linear term in $\phi(ax_i + bx_j)$.

Whence, we get that $\phi(ax_ix_k + bx_jx_k) = 0$ for all $k \in [n]$ which contradicts the assumption that ϕ is an isomorphism. Hence, π is a permutation on $[n]$. Now look at the action of ϕ on the edges-polynomial:

$$\begin{aligned} 0 &= \phi\left(\sum_{\substack{1\le i<j\le n\\(i,j)\in E(G)}} x_ix_j\right) \\ &= \sum_{\substack{1\le i<j\le n\\(i,j)\in E(G)}} \phi(x_i)\phi(x_j) \\ &= \sum_{\substack{1\le i<j\le n\\(i,j)\in E(G)}} c_{i,\pi(i)}c_{j,\pi(j)}x_{\pi(i)}x_{\pi(j)} \end{aligned}$$

Since the above is a zero relation in the ring $R(G')$, we get that the polynomial

$$\sum_{\substack{1\le i<j\le n\\(i,j)\in E(G')}} x_ix_j$$

divides the above. Hence, $(\pi(i), \pi(j)) \in E(G')$ if $(i, j) \in E(G)$.

By symmetry this shows that π is an isomorphism from $G \to G'$.

□

References

[AB03] Manindra Agrawal and Somenath Biswas. Primality and identity testing via chinese remaindering. *JACM: Journal of the ACM*, 50, 2003.

[Agr05] Manindra Agrawal. Proving lower bounds via pseudo-random generators. In R. Ramanujam and Sandeep Sen, editors, *FSTTCS 2005: Foundations of Software Technology and Theoretical Computer Science, 25th International Conference, Hyderabad, India, December 15-18, 2005, Proceedings*, volume 3821 of *Lecture Notes in Computer Science*, pages 92–105. Springer, 2005.

[AKS04] Manindra Agrawal, Neeraj Kayal, and Nitin Saxena. Primes is in p. *Annals of Mathematics*, 160(2):781–793, 2004.

[AM94] Adleman and McCurley. Open problems in number theoretic complexity, II. In *ANTS: 1st International Algorithmic Number Theory Symposium (ANTS)*, 1994.

[BD66] Enrico Bombieri and H. Davenport. On two problems of mordell. *American Journal of Mathematics*, 88:61–70, 1966.

[Ber67] E. R. Berlekamp. Factoring polynomials over finite fields. *Bell System technical Journal*, 46:1853, 1967.

[Ber70] E. R. Berlekamp. Factoring polynomials over large finite fields. *Mathematics of Computation*, 24(111):713–735, July 1970.

[BGry] Mihir Bellare and Shafi Goldwasser. The complexity of decision versus search. *SIAM J. Comput.*, 23(1):97–119, 1994, February.

[BS84] Laszlo Babai and Endre Szemerédi. On the complexity of matrix group problems. In *Proceedings of the 25th Symposium on Foundations of Computer Science (FOCS)*, pages 229–240. IEEE Computer Society Press, 1984.

[CEG83] E. R. Canfield, Paul Erdos, and Andrew Granville. On a problem of oppenhein concerning factorisatio numerorum. *Journal of Number Theory*, 17:1–28, 1983.

[CK00] Zhi-Zhong Chen and Ming-Yang Kao. Reducing randomness via irrational numbers. *SIAM J. Comput.*, 29(4):1247–1256, 2000.

[CM03] A. Cafure and G. Matera. Explicit estimates for the number of solutions of polynomial equation systems over finite fields, February 04 2003.

[CM04] Antonio Cafure and Guillermo Matera. Improved explicit estimates on the number of solutions of equations over a finite field, May 14 2004. Comment: 33 pages.

[Coh70] S. D. Cohen. The distribution of polynomials over finite fields. *Acta Arithmetica*, 17:255–271, 1970.

[Coo71] S. A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing*, pages 151–158, New York, 1971.

[CZ81] D. G. Cantor and H. Zassenhaus. A new algorithm for factoring polynomials over finite fields. *Mathematics of Computation*, 36(154):587–592, 1981.

[DL63] H. Davenport and D.J. Lewis. Notes on congruences (i). *Quaterly Journal of Mathematics Oxford*, 14:51–60, 1963.

[DS05] Zeev Dvir and Amir Shpilka. Locally decodable codes with 2 queries and polynomial identity testing for depth 3 circuits. In *STOC*, pages 592–601, 2005.

[dW06] Christiaan Van de Woestijne. *Deterministic equation solving over finite fields.* PhD thesis, Universiteit Leiden, 2006.

[Gat89] Joachim Von Zur Gathen. Testing permutation polynomials (extended abstract). In *FOCS*, pages 88–92. IEEE, 1989.

[Gat91] Joachim Von Zur Gathen. Tests for permutation polynomials. *SIAM Journal on Computing*, 20(3):591–602, June 1991.

[GK98] Dima Grigoriev and Marek Karpinski. An exponential lower bound for depth 3 arithmetic circuits. In *STOC*, pages 577–582, 1998.

[GKL04] Shuhong Gao, Erich Kaltofen, and Alan G. B. Lauder. Deterministic distinct-degree factorization of polynomials over finite fields. *Journal of Symbolic Computation*, 38(6):1461–1470, December 2004.

[GS99] Venkatesan Guruswami and Madhu Sudan. Improved decoding of reed-solomon and algebraic-geometry codes. *IEEE Transactions on Information Theory*, 45(6):1757–1767, 1999.

[Gur01] 1976-Venkatesan Guruswami. *List decoding of error-correcting codes*. PhD thesis, Massachusetts Institute of Technology, Dept. of Electrical Engineering and Computer Science, 2001.

[Hay67] D. R. Hayes. A geometric approach to permutation polynomials over a finite field. *Duke Mathematics Journal*, 34:293–305, 1967.

[Her75] I. N. Herstein. *Topics in Algebra*. John Wiley & Sons, New York, 2nd edition, 1975.

[HW96] Ming-Deh Huang and Yiu-Chung Wong. Solving systems of polynomial congruences modulo a large prime (extended abstract). In *37th Annual Symposium on Foundations of Computer Science*, pages 115–124, Burlington, Vermont, 14–16 October 1996. IEEE.

[HW99] Ming-Deh A. Huang and Yiu-Chung Wong. Solvability of systems of polynomial congruences modulo a large prime. *Computational Complexity*, 8(3):227–257, 1999.

[HW00] Huang and Wong. Extended hilbert irreducibility and its applications. *ALGORITHMS: Journal of Algorithms*, 37, 2000.

[IK03] Impagliazzo and Kabanets. Derandomizing polynomial identity tests means proving circuit lower bounds. In *STOC: ACM Symposium on Theory of Computing (STOC)*, 2003.

[IW97] Impagliazzo and Wigderson. P = BPP if E requires exponential circuits: Derandomizing the XOR lemma. In *STOC: ACM Symposium on Theory of Computing (STOC)*, 1997.

[Kal82] Erich Kaltofen. A polynomial-time reduction from bivariate to univariate integral polynomial factorization. In *FOCS*, pages 57–64, Chicago, Illinois, 3-5 November 1982. IEEE.

[Kal85] E. Kaltofen. Fast parallel absolute irreducibility testing. *JSC*, 1(1):57–67, March 1985.

[Kar72] R. M. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, pages 85–103. Plenum Press, NY, 1972.

[Kay05] Kayal. Solvability of a system of bivariate polynomial equations over a finite field. In *ICALP: Annual International Colloquium on Automata, Languages and Programming*, 2005.

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

[Kla89] Andrew Klapper. Generalized lowness and highness and probabilistic complexity classes. *Mathematical Systems Theory*, 22(1):37–45, 1989.

[Knu81] D. Knuth. *The Art of Computer Programming, Volume 2: Seminumerical Algorithms*. Addison-Wesley, Reading, MA., 1981.

[KS01] Adam Klivans and Daniel A. Spielman. Randomness efficient identity testing of multivariate polynomials. In *STOC*, pages 216–223, 2001.

[KS05] Neeraj Kayal and Nitin Saxena. On the ring isomorphism and automorphism problems. In *IEEE Conference on Computational Complexity*, pages 2–12. IEEE Computer Society, 2005.

[KS06] Neeraj Kayal and Nitin Saxena. Polynomial identity testing for depth 3 circuits. In *Proceedings of the twenty-first Annual IEEE Conference on Computational Complexity (CCC)*, 2006.

[KST93] J. Köbler, U. Schöning, and J. Torán. *The Graph Isomorphism Problem*. Birkhäuser, 1993.

[Lan94] S. Lang. *Algebra*. Reading (MA): Addison-Wesley, 3. edition, 1994.

[Len91] H. W. Lenstra, Jr. Finding isomorphisms between finite fields. *Mathematics of Computation*, 56(193):329–347, January 1991.

[Len04] Hendrik Lenstra. On rigid non-commutative rings. *Private communication*, 2004.

[LLL82] Lenstra, Lenstra, and Lovasz. Factoring polynomials with rational coefficients. *MATHANN: Mathematische Annalen*, 261, 1982.

[LM83] Lidl and Muller. Permutation polynomials in RSA-cryptosystems. In *CRYPTO: Proceedings of Crypto*, 1983.

[LN94] Rudolf Lidl and Harald Niederreiter. *Introduction to finite fields and their applications.* Cambridge University Press, Cambridge, 1994.

[LV98] Daniel Lewin and Salil P. Vadhan. Checking polynomial identities over any field: Towards a derandomization? In *STOC*, pages 438–447, 1998.

[Mac67] C. R. MacCluer. On a conjecture of davenport and lewis concerning exceptional polynomials. *Acta Arithmetica*, 12:289–299, 1967.

[McD74] Bernard R. McDonald. *Finite Rings with Identity.* Marcel Dekker Inc., 1974.

[MG95] Ma and Von Zur Gathen. The computational complexity of recognizing permutation functions. *CMPCMPL: Computational Complexity*, 5, 1995.

[Mil76] Gary L. Miller. Riemann's hypothesis and tests for primality. *Journal of Computer and System Sciences*, 13(3):300–317, December 1976.

[Nai82] M. Nair. On chebyshev-type inequalities for primes. *American Mathematical Monthly*, 89:126–129, 1982.

[Poo02] Bjorn Poonen. Computing rational points on curves. In M.A. Bennett et al, editor, *Number Theory For The Millenium III*, pages 149–172, 2002.

[Pud94] Pavel Pudlák. Communication in bounded depth circuits. *Combinatorica*, 14(2):203–216, 1994.

[Raz04] Ran Raz. Multi-linear formulas for permanent and determinant are of super-polynomial size. In *Proceedings of the thirty-sixth annual ACM Symposium on Theory of Computing (STOC-04)*, pages 633–641, New York, June 13–15 2004. ACM Press.

[Rei05] Omer Reingold. Undirected ST-connectivity in log-space. In Harold N. Gabow and Ronald Fagin, editors, *Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005*, pages 376–385. ACM, 2005.

[RS01] Ran Raz and Amir Shpilka. Lower bounds for matrix product, in bounded depth circuits with arbitrary gates. In *STOC*, pages 409–418, 2001.

[RS04] Ran Raz and Amir Shpilka. Deterministic polynomial identity testing in non-commutative models. In *IEEE Conference on Computational Complexity*, pages 215–222, 2004.

[Sax06] Nitin Saxena. *Automorphisms of rings and applications to complexity of problems.* PhD thesis, Indian Institute of Technology, Kanpur, 2006.

[Sch74] Wolfgang M. Schmidt. A lower bound for the number of solutions of equations over finite fields. *Journal of Number Theory*, 6:448–480, 1974.

[Sch79] Jacob T. Schwartz. Probabilistic algorithms for verification of polynomial identities (invited). In *EUROSAM*, pages 200–215, 1979.

[Sch80] Jacob T. Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *J. ACM*, 27(4):701–717, 1980.

[Sch88] Uwe Schöning. Graph isomorphism is in the low hierarchy. *Journal of Computer and System Sciences*, 37(3):312–323, December 1988.

[Sha94] Igor R. Shafarevich. *Basic algebraic geometry 1. Varieties in projective space.* Springer-Verlag, Berlin-Heidelberg-New York, 1994.

[Shp92] Shparlinski. A deterministic test for permutation polynomials. *CMPCMPL: Computational Complexity*, 2, 1992.

[SW99] Amir Shpilka and Avi Wigderson. Depth-3 arithmetic formulae over fields of characteristic zero. In *IEEE Conference on Computational Complexity*, pages 87–, 1999.

[Wil68] K. S. Williams. On exceptional polynomials. *Canadian Mathematical Bulletin*, 11:279–282, 1968.

[Zip79] R. Zippel. Probabilistic algorithms for sparse polynomials. In *ISSAC '79: Proc. Int'l. Symp. on Symbolic and Algebraic Computation,* Lecture Notes in Computer Science, Vol. 72. Springer-Verlag, 1979. Zippel discusses probabilistic methods for testing polynomial identities and properties of systems of polynomials.

Index