DISS. ETH NO. 30033

Cryptanalysis by Algebraic Relations

A thesis submitted to attain the degree of

DOCTOR OF SCIENCES
(Dr. sc. ETH Zurich)

presented by

Akin Ünal
M.Sc., Ruprecht-Karls-Universität Heidelberg
born on 24.10.1993

accepted on the recommendation of
Prof. Dr. Dennis Hofheinz, examiner
Prof. Dr. Ueli Maurer, co-examiner
Prof. Dr. Chris Brzuska, co-examiner

2024

Acknowledgments & Danksagungen

Zuallererst und zuallermeist gilt meine aufrechte Anerkennung meinen Eltern, denen—im Gegensatz zu mir—nichts im Leben geschenkt wurde. Meine Eltern kamen als Kinder von Gastarbeitern nach Deutschland und haben sich kaputt gearbeitet, damit meine Schwester und ich ein besseres Leben haben können. Nur Dank der auszehrenden Arbeit und bedingungslosen Liebe meiner Eltern kann ich an diesem Punkt im Leben stehen und dafür schulde ich ihnen meinen endlosen Dank. Allgemeiner gilt meine ungebrochene Solidarität all denjenigen, die auf der Suche nach einem besseren Leben—auf welche Weise auch immer—in fremde Länder reisen.

Meiner Schwester wünsche ich den besten Erfolg und das größte Glück für ihr Leben in Tokyo, das sie vor ein paar Jahren begann. Asya, Du hast noch die beste Zeit vor Dir!

On the academic site, I want to thank everybody with which I had the pleasure to work with. This includes my old colleagues at KIT who are Sven, Michael, Jessica, Kathrin, Jörn, Jiaxin, Julia, Markus, Willi, Valerie, Holger, Frau Manietta, Jeremias, Dingding, Astrid, D.r. Meta, Lisa, Geoffroy, Andy, Mario, Marcel, Brandon, Anna-Louise, Gunnar, Björn, Alex, Thomas, Thomas (other one), Augusto, Killian and many more. It also includes my former and closest colleagues at ETH, which are of course Julia (other one), Nico, Roman, Bogdan, Sebastian, Karen, Lossy “the Trapdoor Squirrel”, Kristina, Michael (other one), Suvradip, Cecilia, Kaan, Yuxi, Xiaoqi and Claudia. Of course, I won’t forget all my good friends at Ueli’s group, Marta, Guilherme, Gianluca, Giovanni, Konstantin, Fabio, David, Chen-Da, Daniel, Jiamin and many more. I may not recall everybody in Kenny’s group, but I have very fond memories of Simon, Varun, Francesca, Kien, Laura, Mia, Jan, Anu, Matteo, Barbara, Kenny (of course) et alias. And outside of our institute I had the pleasure to meet and work with Erkan, Ivy, Simon (other one), Thomas (another other one), Diana, Andrei, Romain and many, many more. I also want to express my gratitude towards my new colleagues at ISTA who invited me over last year and convinced me to stay at their institute. I am looking forward to many fruitful collaborations with Krzysztof, Ben, Ahad, Miguel, Christoph, Charlotte, Guillermo and Anshu.

Finally, I want to express my gratitude towards the examiners of my doctoral committee:

I thank Chris Brzuska for taking the time to read my thesis, coming personally to my exam and introducing me to many more interesting cryptographic questions.

I thank Ueli Maurer whose works I encountered too late, but which inspire me to strive for clean mathematical models of cryptography and cryptanalysis.

Schließlich schulde ich meinen tiefsten Dank meinem Doktorvater, Dennis Hofheinz. Nicht nur hat Dennis mich als frischen Mathematiker in die Kunst der Kryptographie und das Handwerk des wissenschaftlichen Schreibens eingeführt, sondern mit einer Engelsgeduld, die ich bis heute nur bewundern kann, all meine Streiche, all meine Aufstände, all meine Launen und all meine Eskapaden ertragen. Dafür, lieber Dennis, hast Du meinen endlosen Dank!

Reader! Please forgive me if I forgot your name. I had to write these acknowledgments in a hurry, since I forgot some deadlines (again).

Abstract

Algebraic relations are a high-degree generalization of linear relationships. Given an algebraic relation it is possible to predict the outcome of single components of a polynomial equation system or refute points that do not lie in the image of a polynomial map. This work investigates the application of algebraic relations to cryptanalysis, presenting novel insights and algorithms. As a first result, we will show a simple theorem that guarantees the existence of algebraic relations of sublinear degree for overdetermined polynomial equation systems. Equipped with algebraic relations of small degree, we will then show lower bounds for primitives of two cryptographic areas:

Pseudorandom Generators. A pseudorandom generator (PRG) is a deterministic function that stretches a random bit string to a longer bit string, which is supposed to be indistinguishable from uniform randomness. While efficiently computable PRGs do exist, PRGs of low complexity are of special interest, as they are in high demand to construct advanced cryptosystems. However, as we will see, PRGs that are evaluated by polynomials of constant degree or locality admit algebraic relations of low degree. This opens an algebraic attack surface, which yields new attack algorithms of subexponential runtime on lightweight PRGs. In particular, in the setting of polynomial or local PRGs of small polynomial stretch, our algorithms asymptotically surpass all other currently known attack algorithms.

Our insights for algebraic PRGs extend to the popular class of Macaulay matrix-based solving algorithms for polynomial equation systems. We will give the first proof that the overdetermined multivariate quadratic polynomial problem can be solved in the average case over small fields by computing Macaulay matrices up to some sublinear degree.

Functional Encryption. A functional encryption (FE) scheme allows fine-grained access to encrypted data. More specifically, in an FE scheme a master secret key-holder can issue special functional keys that only admit the decryption of evaluations of specific functions on the data of ciphertexts. Since it is known that compact FE implies indistinguishability obfuscation (Bitansky-Vaikuntanatha FOCS 2015, Ananth-Jain CRYPTO 2015), investigating the feasibility of FE schemes is of large interest in cryptographic research.

While FE schemes for linear functions do exist from a plethora of assumptions, it is an open problem to construct lattice-based FE schemes for richer functionalities. This is counterintuitive, as lattice-based hardness assumptions, e.g. learning with errors (Regev STOC 2005), imply strong cryptographic primitives, including fully homomorphic encryption (Brakerski-Vaikuntanathan FOCS 2011). In this work, we will attempt to explain why strong assumptions like learning with errors cannot imply richer FE schemes. Concretely, we will use algebraic relations to prove two lower bounds: if no bit-decomposition is used, we can completely rule out the feasibility of lattice-based function-hiding FE. Further, we can rule out the existence of lattice-based compact FE schemes for strict parameter choices.

Zusammenfassung

Algebraische Abhängigkeiten sind eine Verallgemeinerung von linearen Abhängigkeiten für höhere Grade. Mithilfe einer algebraischen Abhängigkeit ist es möglich, das Ergebnis einer einzelnen Komponente eines polynomiellen Gleichungssystems vorherzusagen, oder Punkte, die nicht im Bild einer polynomiellen Abbildung liegen, zurückzuweisen. Diese Arbeit untersucht die Anwendung algebraischer Abhängigkeiten zur Kryptoanalyse und präsentiert neue Erkenntnisse und Algorithmen. Als ein erstes Resultat werden wir ein simples Theorem zeigen, das die Existenz algebraischer Abhängigkeiten für polynomiell überbestimmte polynomielle Gleichungssystem garantiert. Ausgestattet mit algebraischen Abhängigkeiten kleiner Grade werden wir dann untere Schranken für Primitive aus zwei kryptografischen Gebieten zeigen:

Pseudozufallsgeneratoren. Ein Pseudozufallsgenerator ist eine deterministische Funktion, die einen zufälligen Bitstring zu einem längeren Bitstring streckt, der von uniformen Zufall ununterscheidbar sein soll. Zwar existieren effizient berechenbare Pseudozufallsgeneratoren, allerdings sind Pseudozufallsgeneratoren niedriger Komplexität von besonderem Interesse, da an ihnen ein hoher Bedarf zur Konstruktion fortgeschrittener Kryptosysteme existiert. Allerdings werden wir sehen, dass Pseudozufallsgeneratoren, die von Polynomen konstanten Grades oder Lokalität berechnet werden, algebraischen Abhängigkeiten niedrigen Grades besitzen. Das eröffnet eine algebraische Angriffsfläche, wodurch sich neue Angriffsalgorithmen subexponentieller Laufzeit auf leichtgewichtige Pseudozufallsgeneratoren ergeben. Besonders im Kontext der polynomiellen und lokalen Pseudozufallsgeneratoren kleiner polynomieller Streckung werden unsere Algorithmen asymptotisch alle anderen bisher bekannten Angriffsalgorithmen übertreffen.

Unsere Einsichten für algebraische Pseudozufallsgeneratoren lassen sich übertragen auf die populäre Klasse der auf Macaulaymatrizen basierenden Lösungsalgorithmen für polynomielle Gleichungssysteme. Wir werden den ersten Beweis dafür geben, dass das Problem der überbestimmten multivariaten quadratischen Polynome über kleinen Körpern im Durchschnittsfall durch das Berechnen von Macaulaymatrizen bis zu einem bestimmten sublinearen Grad gelöst werden kann.

Funktionale Verschlüsselung. Ein funktionales Verschlüsselungsschema erlaubt detailgenauen Zugriff auf verschlüsselte Daten. Um genauer zu sein, in einem funktionalen Verschlüsselungsschema kann der Besitzer eines geheimen Hauptschlüssels spezielle funktionale Schlüssel ausstellen, die nur die Entschlüsselung der Auswertungen spezifischer Funktionen an den verschlüsselten Daten erlauben. Da bekannt ist, dass kompakte funktionale Verschlüsselungsschemata Ununterscheidbarkeitsobfuskatoren implizieren (Bitansky-Vaikuntanatha FOCS 2015, Ananth-Jain CRYPTO 2015), ist das Untersuchen der Realisierbarkeit funktionaler Verschlüsselungsschemata von großem Interesse in der kryptografischen Forschung.

Während funktionale Verschlüsselungsschemata für lineare Funktionen von einer Vielzahl von Annahmen existieren, ist es ein offenes Problem gitterbasierte Verschlüsselungsschemata für mächtigere Funktionalitäten zu konstruieren. Dies ist kontraintuitiv, da gitterbasierte Annahmen, wie zum Beispiel Lernen mit Fehlern (Regev STOC 2005), starke kryptografische Primitive, einschließlich voll-homomorpher Verschlüsselung (Brakerski-Vaikuntanathan FOCS 2011), implizieren.

In dieser Arbeit werden wir versuchen zu erklären, warum starke Annahmen wie Lernen mit Fehlern nicht mächtigere funktionale Verschlüsselungsschemata implizieren können. Konkret werden wir algebraische Abhängigkeiten benutzen, um zwei untere Schranken zu beweisen: falls keine Bitzerlegung benutzt wird, können wir die Realisierbarkeit gitterbasierter, funktionsprivater funktionaler Verschlüsselung komplett ausschließen. Ferner, können wir die Existenz gitterbasierter, kompakter funktionaler Verschlüsselungsschemata für strikte Parameterwahlen ausschließen.

Contents

Introduction

Motivation

On the Necessity of Cryptanalysis

The one-time pad, invented by Miller [Mil82] and reinvented by Vernam [Ver26], is a prime example of a symmetric encryption scheme. Its security has been proven by Shannon [Sha49], and a special feature of the one-time pad and its security proof is that its security is indisputable: everybody who assumes the soundness of basic mathematical axioms must acknowledge the security of the one-time pad. Such unconditional security guarantees cannot be given for richer cryptosystems. For example, it is known that the security of public-key encryption schemes, a notion introduced by Diffie and Hellman [DH76], must imply that the class **P** of efficiently decidable languages cannot contain the class **NP** of languages with efficiently verifiable witnesses of membership. However, even under complexity-theoretic conjectures such as $\mathbf{P} \neq \mathbf{NP}$ one can not prove the existence of a secure public-key cryptosystem. Indeed, the first provably secure public-key cryptosystem, given by Rivest, Shamir, and Adleman [RSA78], needs to base its security on a *hardness assumption*, concretely on the assumption that it is intractable to take roots of random numbers modulo the product of two large unknown primes.

This approach led to the generally recognized strategy of proving the security of a cryptographic primitive by assuming the hardness of a specific mathematical problem. Ideally, this yields a win-win situation for cryptographers and mathematicians: either a cryptographic primitive is secure or a successful attack on said primitive can—by a cryptographic reduction—be turned into a breakthrough algorithm for a mathematical problem that has been open for decades or centuries. In cryptographic practice, however, most hardness assumption are based on problems that are relatively young and of low interest for mathematicians. In fact, an attentive observer might notice that each year renowned cryptography conferences publish papers that introduce new cryptographic hardness assumptions that get broken in the very next year. Since cryptographers cannot rely on mathematicians alone to analyse the hardness of the problems underlying their assumptions, the importance of a second discipline had become apparent: the discipline of *cryptanalysis*, which consists of analysing cryptographic primitives and systems and the hardness assumptions on which their security is based. In fact, cryptanalysis is not the companion of cryptography, it is the very backbone of the scientific foundation of cryptography. Without the ongoing cryptanalytic efforts of the research community, we would have no reason to trust the security of novel cryptosystems.

Polynomial Equation Systems and Algebraic Relations

A polynomial equation system is a finite set of polynomials in multiple indeterminates over a ring or field. Given such a system, one usually asks to find a common root or to decide its satisfiability. The problem of solving and deciding the satisfiability of polynomial equation systems has been studied by mathematicians for centuries. They appear frequently in multiple scientific disciplines and engineering tasks [Laz09], and are known to be **NP**-hard.

A lot of tools have been developed to solve polynomial equation systems, e.g., elimination theory, resultants [Mac16], gradient descent methods, Groebner bases [Buc76], homotopy path continuation [AG90] and many more. In this thesis, we will study one of those tools for handling and reasoning about polynomial equation systems: the tool of *algebraic relations*.

An algebraic relation connects multiple polynomials with each other by making the output behaviour of one polynomial dependent on the output behaviours of all other polynomials. Mathematically, it is itself a non-zero polynomial that must always vanish when evaluated at the outputs of the starting polynomials. Hence, algebraic relationships are higher-degree generalizations of linear dependencies. From an algebraic point of view, it is clear that the members of any system of m polynomials over $n < m$ indeterminates cannot act totally independently of each other and must admit an algebraic relationship among them. This is, because the degree of transcendency of the ring generated by the polynomials over their coefficient field cannot exceed the number of variables. However, if the number of polynomials m does not exceed the number of variables n by a lot, then the degree D of the algebraic relation as a polynomial may be exponential in n. We will see that, with larger ratio m/n, the degree D of the algebraic relation decreases. If $m \geq n^{1+e}$ is by a polynomial amount larger than n, then D will turn out to be sublinear in n.

In the cryptanalytic setting, where multiple polynomials may be chosen by a cryptographic primitive that we want to analyse, algebraic relations allow us to test and restrict the behaviour of those polynomials. Let us explain this in more detail:

- **Refutation.** When given a vector of values, we can decide if this vector is a potential output value of the considered system of polynomials. Since the algebraic relation must vanish on each output of the system, it must also vanish on the given vector. If, on the other hand, the given vector is drawn uniformly at random from its corresponding space, then the Schwartz-Zippel lemma bounds the probability of our algebraic relation vanishing on the given vector. This observation will give us a direct algorithm for distinguishing the output of algebraic pseudorandom generators from true randomness.

- **Restriction.** Let us assume we know the output of all but one polynomial of the system evaluated on a secret value. Since the algebraic relation must vanish on the output of all polynomials, we can use it to restrict the possible output values of the one polynomial of the system whose output value we did not learn. Concretely, we can evaluate the algebraic relation on all output values that we know. This yields a univariate polynomial whose degree is bounded by the degree D of the algebraic relation. Since this polynomial must vanish on the output of the one polynomial whose output value we do not know, this unknown output value must be a root of this univariate polynomial of degree D. Hence, we can compute a set of at most D elements in which the undisclosed output value must lie. If D is non-constant, this procedure is not efficient, however, it still gives us an information-theoretic extractor. At this point, it is not apparent how, but we will use this observation to prove lower bounds for compact functional encryption scheme that fit into our model of lattice-based schemes that we will introduce here.

Before we continue with the technical details, we will explain both fields of primitives that we analyse in more detail.

Algebraic Pseudorandom Generators

A *pseudorandom generator* (PRG) is a deterministic function that maps n bits to $m > n$ bits. It guarantees that its output, when evaluated on a uniformly random string of n bits, is indistinguishable from a uniformly random string of m bits. Hence, it stretches sequences of true randomness into longer sequences of pseudorandomness.

Generating randomness is an important aspect of most cryptographic primitives in theory and in reality. In theoretical cryptography, we study PRGs to investigate how well we can imitate randomness with bounded resources. Further, PRGs stand in direct correspondence with one-way functions [HILL99]. In complexity theory, PRGs can be used to derandomize probabilistic algorithms. For example, Impagliazzo and Wigderson [IW97] showed that the existence of *high-end* PRGs implies that the class **P** coincides with the class **BPP** of languages decidable by a PPT algorithm with bounded probability of erring per instance.

While it is generally acknowledged that efficiently computable PRGs do exist, the minimum complexity required for a deterministic function $F : 0, 1^n \to 0, 1^m$ to be a strong PRG with superlinear stretch remains unclear. We will address a specific question: what is the highest level of complexity that a PRG, computed by binary polynomials of constant degree, can deceive? First, we will show that if F is a poly-stretch, i.e., $m \geq n^{1+e}$, then it cannot deceive certain algebraic adversaries. These adversaries operate under subexponential time complexity, specifically 2^{n^δ} for some $\delta < 1$. Hence, the bit security of an algebraic PRG $F : \{0,1\}^n \to \{0,1\}^m$ must be polynomially smaller than the seed size n. Now, algebraic PRGs and one-way functions are interesting from the viewpoint of multivariate cryptography, a potentially post-quantum secure branch of cryptography that is way less studied than its competitor lattice-based cryptography. In particular, with regard to multivariate cryptography, one can extend the notion of PRGs to polynomial functions $F : k^n \to k^m$ over finite fields k. In fact, these objects will be the main focus of the first half of this work, and our results for PRGs over finite fields will be applicable to binary PRGs.

Local PRGs. A well-researched subclass of algebraic PRGs is the class of local PRGs. The notion of local functions has been introduced by Goldreich

[Gol11]: a function $F : \{0,1\}^n \to \{0,1\}^m$ is called *local* if each of its output bits is computed by evaluating a constant-size set of its input bits. The most notorious candidate PRG of constant locality is given by the family of PRGs induced by applying the tri-sum-and predicate

$$P(X_1, X_2, X_3, X_4, X_5) := X_1 \oplus X_2 \oplus X_3 \oplus X_4 \cdot X_5 \tag{1}$$

on random sets of input bits. This candidate has been put forth by Mossel, Shpilka, and Trevisan [MST03] who showed that it has a negligible bias for small superlinear stretches. The very low complexity of local PRGs leads to their use in several advanced protocols. Let us name a few:

- **Two-Party Protocols.** Ishai, Kushilevitz, Ostrovsky, and Sahai [IKOS08] constructed communication protocols that allowed two parties to jointly evaluate a circuit on both parties' data without any party learning a non-trivial amount of the other party's data. The security of their protocols are based on local PRGs of polynomial stretch and oblivious transfer. In the semi-honest model, i.e., both parties do not cheat, the computational complexity of the protocol is linear in the size of the circuit they want to evaluate. In the malicious model, where we allow both parties to cheat, the protocol can be fixed s.t. it is secure again and has a superlinear computational overhead. Applebaum, Damgård, Ishai, Nielsen, and Zichron [ADINZ17] and Applebaum and Konstantini [AK23] extended both protocols for arithmetic circuits by making use of PRGs $F : k^n \to k^m$ of polynomial stretch over finite fields k. Their protocols improve the status quo by retaining linear computational overhead in the semi-honest and malicious model.

- **Usage in Multi-Party Protocols and Fully Homomorphic Encryption.** The evaluation of local functions does not need to perform a lot of multiplications or additions and, hence, tends to be very friendly towards fully homomorphic encryption and multiparty protocols. In particular in the context of fully homomorphic encryption, a smaller depth of functions evaluated on ciphertexts allows a smaller modulus-to-noise ratio of the underlying learning with errors assumption, which leads to a better security.

- **Indistinguishability Obfuscation.** Building up on a long line of work, Jain, Lin, and Sahai [JLS21; JLS22] devised an indistinguishability obfuscation candidate whose security could be proven upon three established assumptions of cryptography: learning parity with noise, bilinear groups over elliptic curves and local PRGs of polynomial stretch. However, to instantiate an obfuscation scheme they need to assume the *subexponential* security of the involved assumptions, i.e., their assumptions need to resist a larger class of adversaries that may perform a superpolynomial number of computational steps. As we will see later, almost all obfuscation candidates need to base their security on subexponentially secure assumption. This fact additionally raises the interest into attack algorithms of superpolynomial and subexponential time complexity. Hence, we will tend in Chapter 1 to the following question:

What is the maximum security a PRG $F : \{0,1\}^n \to \{0,1\}^{n^{1+e}}$ of polynomial stretch and constant degree or locality d can offer?

We will come back to obfuscation when discussing functional encryption, since both topics are intertwined.

Pairings and Learning With Errors

Before we continue with functional encryption, it makes sense to introduce some hardness assumptions based on pairings and on lattices, since both notions will be mentioned regularly when discussing the hardships of constructing functional encryption schemes.

Pairing-Based Cryptography. In group-based cryptography, the existence of a (usually) cyclic group of exponential prime cardinality is assumed where typical group operations like products and inversions of group elements can be efficiently performed. Additionally, one assumes or requires that the neutral element and a generator of the group are known and that each group element has a unique binary representation of succinct size.

Regarding the hardness, there are a lot of different problems over the group that one can assume to be intractable. The most well-researched one is the discrete logarithm problem that asks to find $x \in \mathbb{Z}$ s.t. $g^x = h$ for two uniformly random group elements g, h. The computational Diffie-Hellman problem asks to multiply the *exponents* of uniformly random group elements, and the decisional Diffie-Hellman problem asks to distinguish the exponentiation product of two given random group elements from a third random group element.

Initial instantiations of such groups were given by the unit groups of large fields, however, it turned out that groups of points over elliptic curves yield better security. An idealization of such groups is given by generic group models, which were introduced by Shoup [Sho97] and Maurer [Mau05].

Now, a *pairing* over cyclic groups G_1, G_2, G_T of cardinality $p \in \Theta(2^\lambda)$ is given by a bilinear map

$$e : G_1 \times G_2 \longrightarrow G_T \tag{2}$$

that maps designated generators g_1 and g_2 of G_1 and G_2, respectively, to a generator g_T of G_T. In the cryptographic setting, concrete instantiations of pairings are given by the Weil and Tate pairing over groups G_1, G_2 over elliptic curves [MVO91; FR94; FMR99; Ver01; JN03; Jou04; KM05; GPS08]. To make cryptographic use of the ensemble (e, G_1, G_2, G_T), one needs to assume the hardness of a problem that involves all participating domains. An example is the computational bilinear Diffie-Hellman assumption [BF01] that states that it is hard to compute g_T^{xyz} when given access to

$$(g_1, g_2, g_1^x, g_1^y, g_2^z) \tag{3}$$

for uniformly random exponents $x, y, z \leftarrow \mathbb{Z}_p$.

While group-based cryptography can make use of $\mathbb{Z}_p$-modules with black-box access, pairings take the field one level higher and allow for the evaluation of degree-2 functions over those modules. This opened the door for a huge domain[1] of applications in cryptography, which go far beyond functional encryption.

[1] There are efforts to find post-quantum alternatives for pairings. However, as we will see in this work, finding a lattice-based alternative proves to be challenging. Hence, in a not so far future, the looming threat of quantum-computers may cause a hard-to-recover loss of a lot of cryptosystems that can yet only be constructed from pairings.

Lattice-Based Cryptography. A lattice is a discrete additive subgroup of the euclidean space $\mathbb{R}^n$. Problems surrounding lattices—e.g., finding shortest vectors—have been studied for decades and are notoriously hard to solve. A first hardness assumption based on lattice problems was provided by Ajtai [Ajt96] in form of the short integer solution problem. This problem enjoys a worst-case to average-case reduction from well-studied lattice problems, which reinforces the general trust in its hardness.

A more popular lattice-based assumption has been put forth by Regev [Reg05] in the form of *learning with errors* (LWE). The LWE assumption states the indistinguishability of the distributions

$$(A, As + e \bmod q) \qquad\qquad \text{and} \qquad\qquad (A, b) \tag{4}$$

for a uniformly random matrix $A \leftarrow \mathbb{Z}_q^{m \times n}$, uniformly random vectors $s \leftarrow \mathbb{Z}_q^n$, $b \leftarrow \mathbb{Z}_q^m$ and a noise vector $e \leftarrow \chi^m$, whose entries stem from a bounded distribution over $\mathbb{Z}$ of sufficient entropy. Usually, the discrete Gaussian distribution is taken for χ. Note that this problem is parametrized by n, i.e., its hardness grows by its dimensions. However, the number m of samples is usually unbounded, as having many samples does not necessarily help in solving the problem. Like short integer solution, LWE also enjoys a worst-case to average-case reduction from notoriously hard lattice problems. Extensions of LWE are given by Ring-LWE [LPR10] and Module-LWE [LS15], which we will not discuss here.

Not only is LWE a very versatile assumption that helped to instantiate multiple ABE and FE schemes, it is also the only standard assumption yet (besides its relatives Ring-LWE and Module-LWE) from which fully-homomorphic encryption, a notion put forth by Gentry [Gen09], could be instantiated [BV11]. However, while LWE is a very powerful assumption, there are certain cryptographic primitives that cannot be instantiated by it and its relatives, but for which instantiations from pairings are known. In the following, we will introduce and discuss those primitives. Investigating why certain functionalities cannot be supported by LWE is of particular interest, since this helps us to understand the boundaries of LWE and lattice-based assumptions, in general. Let us ask the following question here:

Are there inherent boundaries that separate the capabilities of LWE from the capabilities of pairing-based assumptions?

The lower bounds that we will work out in Chapter 2 are an attempt towards understanding this question.

Lattice-Based Functional Encryption

A *functional encryption* (FE) scheme allows fine-grained access on encrypted data. Concretely, a master secret key-holder can issue secret keys that allow a user to only learn the evaluation value of a specific function on the messages of received ciphertexts. The notion of functional encryption has been recently introduced by [ONe10; BSW11] as the culmination of ongoing generalizations of message encryption schemes. Let us give a short overview of this development; a more elaborate survey can be found in the doctoral thesis of Gay [Gay19].

In *identity-based encryption* (IBE) schemes—introduced by Shamir [Sha84]—a sender may specify the identity of the receiver while using a global public key for encryption. The receiver holds a secret key for its identity—issued by a master secret key-holder—that allows the receiver to only decrypt messages which are addressed to its identity. First IBE schemes have been given by Boneh and Franklin [BF01] and Cocks [Coc01]. In hierarchical IBE schemes, introduced by Horwitz and Lynn [HL02] and Gentry and Silverberg [GS02], the sender may specify a subdomain of identities that may decrypt its ciphertext. A major generalization has been given by the fuzzy IBE scheme of Sahai and Waters [SW05]. It allows the decryption of a ciphertext if the Hamming distance between the designated identity of the ciphertext and the identity of the secret key is small enough. This gives us the first *attribute-based encryption* (ABE) scheme. In an attribute-based encryption scheme, ciphertexts $\mathsf{ct}_{x,a}$ are endowed with public attributes[2] a, while secret keys sk_ϕ are specified by public predicates ϕ. If the predicate ϕ accepts the attribute a, then the decryption of $\mathsf{ct}_{x,a}$ with sk_ϕ succeeds and yields the message x. A more powerful ABE scheme has been given by Goyal, Pandey, Sahai, and Waters [GPSW06], and ABE schemes that support any circuit as policy have been given by Garg, Gentry, Halevi, Sahai, and Waters [GGHSW13] and Gorbunov, Vaikuntanathan, and Wee [GVW13]. From this point on, one can increase the security even further by hiding additionally the attributes of the ciphertexts. This leads to the notion of *predicate encryption* (PE) schemes, introduced by Boneh and Waters [BW07] and Katz, Sahai, and Waters [KSW08].

Let us finally turn to the notion of functional encryption: the first FE schemes for *bounded collusion* security have been given by Sahai and Seyalioglu [SS10], Gorbunov, Vaikuntanathan, and Wee [GVW12], and Goldwasser, Kalai, Popa, Vaikuntanathan, and Zeldovich [GKPVZ13]. The function space of these schemes supports any circuit, however security is only guaranteed as long as an adversary receives an a priori fixed number of secret keys. Constructing FE schemes that stay secure for an unbounded number of collusions is hard (we will explain later why). Up to now, only for two classes of functionality, FE schemes that support an unbounded number of secret keys could be constructed from standard assumptions[3]: so-called *inner-product encryption* (IPE) schemes have been given by Agrawal, Freeman, and Vaikuntanathan [AFV11], Abdalla, Bourse, De Caro, and Pointcheval [ABDP15], and Agrawal, Libert, and Stehlé [ALS16] based on the security of elliptic curves and learning with errors. IPE schemes support linear functions, i.e., messages are vectors and at decryption the receiver of a ciphertext only learns a linear combination of the entries of the message. A richer functionality is given by *quadratic functional encryption* schemes. In these schemes, the functions evaluated by secret keys are allowed to be degree-2 functions of the coordinates of messages. First FE schemes for quadratic functions have been given by Ananth and Sahai [AS17], Lin [Lin17], and Baltico, Catalano, Fiore, and Gay [BCFG17].

There is a plethora of extensions for the security and power of FE schemes, which we will not discuss here. For example, one can include more roles into

[2] One can reverse the roles such that ciphertexts carry a policy-predicate ϕ and secret keys can only decrypt if they contain an accepted attribute a. This leads to the notion of ciphertext-policy attribute-based encryption.

[3] The FE scheme of [JLS22] supports richer functionalities. Its security is based on the polynomial security of local PRGs, learning parity with noise over fields of exponential size and pairings. It is up to discussion if the first two of those assumptions are considered to be standard. Hence, we will not investigate this FE scheme here.

the security notions of FE, which leads to multi-input FE and multi-client FE [Gol+14; AGRW17; ACFGU18], decentralized multi-client FE [CDGPP18], multi-party FE [AGT21], partially-hiding FE [AJLMS19; Wee20; GJLS21; JLMS19] and many more.

From here on, we will instead discuss two other features FE schemes can have: function-privacy and compactness.

Function-Hiding Functional Encryption. A *function-hiding*[4] functional encryption (FHFE) scheme has the special property that secret keys hide the function they evaluate. Hence, an adversary who is given a ciphertext for a message x and a secret key for a function f learns nothing about x and f except of the function value $f(x)$. The notion has been put forth by Shen, Shi, and Waters [SSW09] and Boneh, Raghunathan, and Segev [BRS13a; BRS13b]. Up to now, function-hiding FE schemes whose security can be reduced to standard assumptions can only support inner product encryption, i.e., linear functionality [BJK15; DDM16; BCFG17; Lin17; ACFGU18]. Further, all of those schemes base their security on pairings. In fact, while there are lattice-based IPEs known, there is no function-hiding FE scheme that can provably reduce its security to LWE or one of its relatives. This is noteworthy, since function-hiding IPE can, in some sense, be seen as a pairing-like construction: we have two values which are hidden by their ciphertext and secret key and which we can multiply once. Like in a pairing, we can then check if the result of the multiplication is zero. Hence, a lattice-based function-hiding IPE scheme would appear similar to what could be termed as a lattice-based pairing or multiplication scheme, to put it simply. Now, while LWE is very homomorphic in nature and allows for a lot of linear operations, it seems that it does not allow for even one multiplication with opening, afterwards. This poses a very interesting paradox and raises the following question:

Which mathematical barriers stop us from constructing lattice-based function-hiding functional encryption?

We will tend to this question in Chapter 2.

Compact Functional Encryption. Quadratic polynomials are the richest class of functionality that is non-trivially supported by an FE scheme based on standard assumptions. Of course, by relinearization, one can convert any degree-d polynomial over n variables into a linear function over $\binom{n+d}{d}$ variables. This procedure does trivially yield FE schemes for degree-d polynomials. However, we do not consider such schemes when we talk about FE for polynomials, since those schemes are not compact in the following sense:

For $d > 1$, an FE scheme FE that supports an unbounded number of secret keys for evaluating degree-d polynomials over n variables is called *compact* or *succinct* if there is a constant $e > 0$ s.t. we have for the bit-size of each ciphertext ct of FE

$$\#\mathsf{ct} \in O(n^{d-e}). \tag{5}$$

[4] FHFE schemes are sometimes called *function-private* instead of function-hiding.

However, if FE only supports a bounded[5] number Q of secret keys it may hand out to an adversary before security breaks, it makes sense to call FE compact or succinct if the bit-size of its ciphertexts ct is sublinear in the number Q of collusions it supports, i.e., if we have

$$\#\mathsf{ct} \in O(Q^{1-e}) \tag{6}$$

for a constant $e > 0$. Ananth and Vaikuntanathan [AV19] constructed optimal non-compact FE schemes from minimal assumptions. The security of their public-key FE scheme is based on the existence of public-key encryption, and the security of their private-key FE scheme assumes the existence of one-way functions. Both schemes can support an a priori bounded number of Q collusions and the ciphertexts of both schemes grow linear in Q. As far as it concerns non-compact FE, this is optimal, since any substantial improvement of the ciphertext size would imply indistinguishability obfuscation, as we will see later.

Constructing compact FE is notoriously hard. In fact, only for the functionality of quadratic polynomials it was possible to construct compact FE schemes from standard assumptions [AS17; Lin17; BCFG17; Gay20]. Remarkably, the security of all of those constructions is based on pairings. Again, we meet a barrier for the capabilities of lattice-based schemes, this time it seems that it is impossible to make the product of numbers hidden in an LWE ciphertexts publicly accessible without revealing the single factors in the ciphertext. Note that there also might be "higher" reasons for the hardness of constructing lattice-based compact quadratic encryption: if there would be a compact FE scheme for degree-2 polynomials, whose encryption and decryption procedures only make use of arithmetic operations over a large field $\mathbb{Z}_p$, one might hope to endow this scheme with additional pairing techniques to reach a compact FE scheme for degree-3 polynomials. The jump from 2 to 3 makes a huge difference: a line of work [Lin16; LV16; AS17; Lin17; LT17] could show that trilinear maps together with LWE and, so-called, 3-block local PRGs imply indistinguishability obfuscation. While the subexponential security of 3-block local PRGs is not a standard assumption, it is plausible to assume. Now, if one would be able to attain a degree-3 FE scheme from a lattice-based degree-2 FE scheme and pairings, it could be possible to turn this FE scheme into something similar to a trilinear map, which may give a fairly secure obfuscation scheme. While this is not a strictly formal argument, it gives us some intuition why compact degree-2 FE from LWE is highly desirable. Again, we ask:

Which mathematical barriers stop us from constructing lattice-based compact functional encryption?

We will also address this question in Chapter 2. However, before we continue, let us make here a quick connection between compactness and algebraic relations. Let us assume, that we have a compact FE scheme and let $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$ be a list

[5] The different definitions for bounded and unbounded FE may seem confusing, however, we might reunite them as follows: note that in an FE scheme for degree-d polynomials over n variables, it does not make sense for an adversary to ask for more than $\binom{n+d}{d} - 1$ secret keys. That is, since the space of degree-d polynomials has a basis of $\binom{n+d}{d}$ monomials. Hence, once the adversary knows a sufficient basis of secret keys, it can construct secret keys for additional functions by itself. This motivates to interpret FE schemes for degree-d polynomials as $(\binom{n+d}{d} - 1)$-bounded.

of secret keys for Q different functions. Let each ciphertext of FE be a vector in $\mathbb{Z}_q^m$ and, because of compactness, we can assume that we have

$$m \in O(Q^{1-e}). \tag{7}$$

Let us additionally assume that each secret key sk_i is a polynomial over the m entries of ciphertexts of degree d. This implies that $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$ is a collection of $Q \in \Theta(m^{1+\frac{e}{1-e}})$ polynomials over much fewer variables. We will show that in this case there is an algebraic relation h whose degree is sublinear in m and which relates the keys $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$ in the following way

$$\forall \mathsf{ct}: \quad h(\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_Q(\mathsf{ct})) = 0. \tag{8}$$

As explained before, we can use h for restriction, i.e., we can restrict the evaluation of $\mathsf{sk}_Q(\mathsf{ct})$ at a ciphertext ct to $\leq \deg h$ possible values whenever we know the evaluations $\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_{Q-1}(\mathsf{ct})$. This gives us the following attack vector: we choose functions $f_1, \ldots, f_Q$ and messages $x_0 \neq x_1$ s.t.

$$\forall j \in [Q-1]: \; f_j(x_0) = 0, \qquad f_j(x_1) = 0, \tag{9}$$

$$f_Q(x_0) = 0, \qquad f_Q(x_1) = 1. \tag{10}$$

Now, when given the challenge ciphertext $\mathsf{ct} = \mathsf{Enc}(\mathsf{msk}, x_b)$ for unknown $b \leftarrow \{0,1\}$, the secret keys $\mathsf{sk}_1, \ldots, \mathsf{sk}_{Q-1}$ will formally not help us at extracting b. However, the *useless* secret keys $\mathsf{sk}_1, \ldots, \mathsf{sk}_{Q-1}$ are related to the useful secret key sk_Q. By using this relation, we can restrict the possible outcomes of $\mathsf{sk}_Q(\mathsf{ct})$, which gives us a non-negligible advantage in the IND-CPA security game of FE. We will explain this technique in greater detail in the technical overview.

Interplay with Obfuscation. To complete our understanding of the relevance of compact FE, let us recapitulate here some results on indistinguishable obfuscation:

Barak, Goldreich, Impagliazzo, Rudich, Sahai, Vadhan, and Yang [Bar+01] showed that the strong notion of virtual black-box obfuscation is impossible to achieve. However, they left the feasibility of indistinguishability obfuscation open. An *indistinguishability obfuscation* (iO) scheme is an algorithm that maps circuits to circuits of the same functionality. Its security guarantee dictates that the obfuscations of two circuits of the same size and functionality are indistinguishable. Hence, an iO scheme does not necessarily hide the behaviour of a circuit, but it can hide the implementation details of it.

It has turned out that indistinguishability obfuscation is extremely useful. In fact, a lot of primitives, which were not reached yet from other security assumptions, could be reached under iO. We list the results of [HMS07; HMS10; SW14; AH18; HU19; AHK20] as examples. Unfortunately, trustable iO candidates are hard to construct. The first iO candidates were given by candidate constructions of multilinear maps [CLT13; GGH13; Gar+13; AGIS14; BR14; BGKPS14; BWZ14; CLT15; GGH15; Bra+15; CHLRS15; BMSZ16; CFLMR16; MSZ16; PST14; DGGMM18], which can be seen as degree-n pairings. A long line of work aimed at reducing the degree of the multilinear maps and managed to bring it down to pairings while assuming the strong security of learning parity with noise over large fields and binary local PRGs [Lin16; LV16; AS17; Lin17; LT17; AJLMS19; JLMS19; GJLS21; JLS21; JLS22]. A second, shorter line of work could construct iO by the approach of inefficient obfuscation [LPST16; BDGM20; GP21; BDGM22]. However, here the security is ultimately based on an assumption of unbounded size over the fully homomorphic encryption scheme of Gentry, Sahai, and Waters [GSW13]. Further, there are a lot of adhoc based iO constructions whose security is harder to verify [GJK18; Agr19; AP20; Bar+20; YCY22].

Now, as we mentioned, there is an important interplay between compact FE and iO. In fact, Bitansky and Vaikuntanathan [BV15] and Ananth and Jain [AJ15] showed that subexponentially secure compact[6] public-key FE (for general circuits) implies iO, and later Kitagawa, Nishimaki, and Tanaka [KNT18] showed that secret-key FE is sufficient. On the other hand, one can construct functional encryption [Gar+13; Wat15] and multilinear maps from iO [AFHLP16; FHHL18; Alb+20] plus one-way functions. Hence, we can see a weak equivalence here: subexponentially secure compact FE implies iO+OWF and multilinear maps. Vice versa, iO+OWF and multilinear maps both imply compact FE (where iO+OWF denotes the combination of indistinguishability obfuscation and one-way functions).

The Lattice-Based FE Model. Finally, let us explain how we plan to tend to the questions raised above. Our aim is to prove mathematical lower bound results for lattice-based function-hiding and compact FE. Note that we cannot prove the impossibility of function-hiding and compact FE in general, since provably secure instantiations of such primitives exist from pairing-based assumptions. Hence, given a (private-key) functional encryption scheme $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$, we need to restrict the capabilities of its algorithms at some points. To do so, we observe and distil common behaviour and design choices of FE, ABE, IBE and other encryption schemes whose security is provably reduced to the hardness of learning with errors. Concretely, we can collect two restrictions that are common under lattice-based FE schemes:

1. Almost all lattice-based IBE and FE schemes[7] support so-called *linear* or *noisy decryption*. This means, given a ciphertext $\mathsf{ct} \in \mathbb{Z}_q^m$ and a secret key $\mathsf{sk} \in \mathbb{Z}_q^m$, the decryption algorithm computes and outputs
$$\left\lceil \frac{\langle \mathsf{ct} \mid \mathsf{sk} \rangle}{\lceil q/p \rceil} \right\rfloor \in \{0, \ldots, p-1\}. \tag{11}$$
I.e., it computes the inner product of the vectors ct and sk modulo q, divides the result over the integers by $\lceil q/p \rceil$ (where $p < q$ denotes the modulus of the message space) and, finally, rounds the result of the division to the nearest integer in $\{0, \ldots, p-1\}$. As has been pointed out in [BDGM19], even fully homomorphic encryption schemes support linear decryption. Of course, there are some schemes where—instead of rounding—the result $\langle \mathsf{ct} \mid \mathsf{sk} \rangle \in \mathbb{Z}_q$ of the scalar product is reduced modulo p to a number in $\{0, \ldots, p-1\}$. However, this decryption approach is equivalent to rounding, as we will show in Lemma 53 in the technical

[6] The notion of compactness resp. succinctness is stricter in those works and requires that the encryption time is upper bounded. For simplicity, we will ignore this here.

[7] Let us give [ABB10; CHKP10; ABDP15; ALS16] as examples. Additionally, the decryption procedure of the ABE scheme in [GVW13] is of constant depth if we restrict the scheme to boolean formulas of constant depth.

part of this work. Note that it is impossible to have a compact FE scheme for degree-2 functions where decryption only computes a scalar product between ciphertexts and secret keys. Hence, we will relax the equality in Eq. (11) and include secret keys sk of FE that are of constant degree over m variables.

Concretely, we will require that a lattice-based FE scheme has *constant decryption* depth, i.e., each ciphertext ct is a vector in $\mathbb{Z}_q^m$, each secret key sk is a polynomial in $\mathbb{Z}_q[C_1, \dots, C_m]$ of constant degree and the decryption algorithm works by

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = \left\lceil \frac{\mathsf{sk}(\mathsf{ct})}{\lceil q/p \rceil} \right\rfloor \in \{0, \dots, p-1\}, \tag{12}$$

i.e., Dec evaluates the polynomial sk at ct, divides the result over the integers by $\lceil q/p \rceil$ and rounds the result down to the nearest integer in $\{0, \dots, p-1\}$.

There are of course some exceptions to the rule of noisy constant depth decryption. The FE schemes in [AR17; AP20] apply arithmetic reduction modulo smaller primes at least twice at reduction. Unfortunately, double arithmetic reduction is not equivalent to rounding, and it is an open question how the techniques in this work can be extended to those schemes. Further, as prime examples that avoid the decryption rule here completely, we give the ABE scheme in [Bon+14] and the predicate encryption scheme in [GVW15]. Both schemes make use of a fully homomorphic encryption scheme. At decryption, they evaluate circuits homomorphically at ciphertexts of the fully homomorphic encryption scheme, which makes bit decompositions of ciphertexts inevitable.

2. Additionally, we note that all mentioned lattice-based IBE, ABE and FE schemes possess a certain *offline/online* structure at encryption. I.e., the encryption algorithm Enc has a subroutine $\mathsf{Enc}_{\mathsf{off}}$ that only sees the public-key resp. master secret key at encryption. This offline algorithm $\mathsf{Enc}_{\mathsf{off}}$ then produces randomness $r_1, \dots, r_m$ which Enc uses linearly to encrypt a message $x \in \{0, \dots, p-1\}$. This leads us to require the following:

   For Enc, there is an offline algorithm $\mathsf{Enc}_{\mathsf{off}}$, s.t. Enc on input a master secret key msk and a message $x \in \{0, \dots, p-1\}^n$ proceeds as follows: it samples a tuple $(r_1, \dots, r_m) \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$ of m polynomials of constant degree over n variables, without looking at the message x. Then, it computes the ciphertext ct of x by evaluating the polynomials $r_1, \dots, r_m$ at x modulo q.

We can now precisely state the class of FE schemes that we will study in this work:

Informal Definition 1 (Lattice-Based Functional Encryption)**.** Let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be a functional encryption scheme for messages in $\{0, \dots, p-1\}^n$. We call FE **lattice-based**, if the following requirements are met:

1. There is a prime modulus $q > p$ and an algorithm $\mathsf{Enc}_{\mathsf{off}}$ that on input msk outputs m polynomials $r_1, \dots, r_m \in \mathbb{Z}_q[X_1, \dots, X_n]$ of constant degree.

On input msk and $x \in \{0, \ldots, p-1\}^n$, Enc samples $(r_1, \ldots, r_m) \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$ and outputs the ciphertext

$$\mathsf{ct} := (r_1(x) \bmod q, \ldots, r_m(x) \bmod q) \in \mathbb{Z}_q^m. \tag{13}$$

2. Each secret key output by KeyGen is a polynomial in $\mathbb{Z}_q[C_1, \ldots, C_m]$ of constant degree.
3. Dec on input sk and ct computes and outputs

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = \left\lceil \frac{\mathsf{sk}(\mathsf{ct})}{\lceil q/p \rceil} \right\rfloor \in \{0, \ldots, p-1\}. \tag{14}$$

My Contributions

In this thesis, we will rework the results of [Üna23c; Üna23a; Üna20; TÜ23] and try to improve, extend and generalize them as far as reasonably possible. The merits of this work are of cryptanalytic nature: in the context of algebraic PRGs, we will provably decrease the security of PRGs that are reasonable candidates for cryptosystems in the real world. In the context of lattice-based functional encryption, we will give general attacks on large classes of potential FE schemes, which shows that typical lattice-based approaches are not sufficient to construct compact or function-hiding FE from LWE.

We pay for all of our results directly out of our pockets, i.e., we will prove each result of this work without relying on any assumptions or conjectures. An additional advantage of our results here is that we analyse whole classes of schemes instead of specific schemes. This gives our results a general relevance in the field of low-weight PRGs and lattice-based FE.

Improving Baseline Distinguishers for Algebraic PRGs

In Chapter 1, we will revisit in a streamlined fashion my work in [Üna23c; Üna23a]. A first important result will be to guarantee the existence of algebraic relationships of small degree:

Informal Theorem 1. *Fix $d \in \mathbb{N}$ and let $m \geq n^{1+e}$ for $e > 0$ constant. There is a function*

$$D \in \Theta(n^{1-e/(d-1)}), \tag{15}$$

s.t. for each collection $f_1, \ldots, f_m \in k[X_1, \ldots, X_n]$ of polynomials of degree $\leq d$, there is a polynomial $h \in k[Y_1, \ldots, Y_m]$ s.t.

$$h \neq 0, \qquad h(f_1(X), \ldots, f_m(X)) = 0, \qquad \textit{and} \qquad \deg h \leq D. \tag{16}$$

This leads directly to a series of new subexponential attacks on PRGs of polynomial stretch and constant degree.

Informal Theorem 2. *Let k be a finite field and $F : k^n \to k^m$ a PRG of stretch $m \geq n^{1+e}$ and degree d.*

1. *There is an adversary $\mathcal{A}$ that distinguishes between output values $F(x)$, $x \leftarrow k^n$, and uniformly random points $y \leftarrow k^m$ with an advantage of*
$$\mathsf{adv}_F^{\mathsf{PRG}}(\mathcal{A}) \geq 1 - O\left(\frac{n^{1-e/(d-1)}}{\#k}\right) \tag{17}$$
and a time complexity of $n^{O(n^{1-e/(d-1)})}$.

2. *There is an adversary $\mathcal{A}_{\mathsf{red}}$ against the pseudorandomness of F with an advantage of*
$$\mathsf{adv}_F^{\mathsf{PRG}}(\mathcal{A}_{\mathsf{red}}) \geq (\#k)^{-n^{1-e/(d-1)}} \tag{18}$$
and a time complexity of $n^{O(n^{1-e/(d-1)})}$.

3. *There is an adversary $\mathcal{A}_{\mathsf{ext}}$ against the pseudorandomness of F with an advantage of*
$$\mathsf{adv}_F^{\mathsf{PRG}}(\mathcal{A}_{\mathsf{ext}}) \geq 1 - O\left(\frac{\log(n)^{1/(d-1)}}{n^{e/(d-1)}}\right) \tag{19}$$
and a time complexity of $n^{O(\log(n)^{1/(d-1)} \cdot n^{1-e/(d-1)})}$.

The attacks of Informal Theorem 2 are the first subexponential-time distinguishers on constant-degree PRGs with provably high advantage. In the context of *local* PRGs $F : \{0,1\}^n \to \{0,1\}^m$, where each output bit of the PRG depends on $\ell \in O(1)$ input bits, already a lot of subexponential attacks were known [App13]. However, the attacks presented here are the first ones whose time complexity only depends on the algebraic degree d of the PRG and work for any polynomial stretch $m \geq n^{1+e}$. Hence, for small polynomial stretches $e \in \left(0, \frac{\lfloor 2\ell/3 \rfloor}{2}\right)$, these attacks asymptotically outperform all other known attacks.

In Table 1, we give an exemplary overview of the performance of those new attacks on PRGs $F : \{0,1\}^n \to \{0,1\}^m$ computed by dense degree-2 polynomials, and compare their online bit complexity with the bit complexity of a practical attack by Dinur [Din21a] based on the *polynomial method*.

Solving Polynomial Equations in the Average Case. By a generic search-to-decision reduction, we can use our PRG distinguishers to solve overdetermined polynomial equation systems in the average case:

Informal Theorem 3. *Let k be a field of size q and $m \geq n^{1+e}$. Let $F : k^n \to k^m$ be a system of m uniformly random polynomials over k of degree d. There is an algorithm $\mathcal{B}$ with time complexity in*
$$q \cdot n^{O(\log(n)^{1/(d-1)} \cdot n^{1-e/(d-1)})} \tag{20}$$
that inverts $F(x)$ with high probability. Concretely, it fulfils for each $x \in k^n$
$$\Pr_F\left[\mathcal{B}(F, F(x)) = x\right] \geq 1 - o(1). \tag{21}$$

We will see in Section 1.4 that F does not need to be completely random. It suffices for its linear part to be uniformly random. As far as I know, Informal Theorem 3 gives the first subexponential-time algorithm for solving the poly-stretch multivariate quadratic search problem in the average case over small fields with provably high advantage.

Seed n	Output m	D_{red}	T_{red}	D_{ext}	$[\overline{k} : \mathbb{Z}_2]$	T_{ext}	T_{pm}
128	1448	7	2^{64}	24	5	2^{132}	2^{118}
256	4096	10	2^{102}	44	6	2^{250}	2^{225}
512	11585	15	2^{166}	76	7	2^{462}	2^{435}
1024	32768	21	2^{254}	130	8	2^{850}	2^{855}
2048	92682	30	2^{392}	199	8	2^{1469}	2^{1691}
4096	262144	43	2^{604}	329	9	2^{2619}	2^{3362}
8192	741455	61	2^{917}	535	10	2^{4596}	2^{6702}
16384	2097152	86	2^{1379}	786	10	2^{7486}	2^{13381}
32768	5931642	122	2^{2078}	1250	11	2^{12762}	2^{26736}
65536	16777216	173	2^{3118}	1808	11	2^{20200}	2^{53444}
131072	47453133	245	2^{4659}	2825	12	2^{33618}	2^{106858}
262144	134217728	347	2^{6944}	4372	13	2^{55317}	2^{213683}
524288	379625062	491	2^{10316}	6245	13	2^{85162}	2^{427333}
1048576	1073741824	695	2^{15295}	9565	14	2^{137868}	2^{854629}

Table 1: We list here bit complexities of attacks from Informal Theorem 2 on a PRG $F : \{0,1\}^n \to \{0,1\}^m$ of degree 2 over $\mathbb{Z}_2$ where $m = \lceil n^{1.5} \rfloor$. The first two columns list n and m. D_{red} upper bounds the degree of a reduced algebraic relationship h of F, and $T_{\mathsf{red}} = D_{\mathsf{red}} \cdot \binom{m}{D_{\mathsf{red}}}$ gives an upper bound for the online bit complexity of the resulting attack on F (this means we assume that we know F in advance and can compute h in a preprocessing phase). Note that we can guarantee for this reduced attack only an advantage of $\geq 2^{-D_{\mathsf{red}}}$. D_{ext} gives a strict upper bound for the degree of an algebraic relationship of F over an extension field $\overline{k}$ of $\mathbb{Z}_2$. An upper bound for the online bit complexity of the corresponding attack is given by $T_{\mathsf{ext}} = \binom{D_{\mathsf{ext}}+m'}{m'} \cdot [\overline{k} : \mathbb{Z}_2] \cdot (1 + 3 \cdot [\overline{k} : \mathbb{Z}_2]^2 \cdot D_{\mathsf{ext}})$, for $m' = \left\lfloor \frac{m}{[\overline{k}:\mathbb{Z}_2]} \right\rfloor$. The degree of the field extension $\mathbb{Z}_2 \subset \overline{k}$ has been chosen s.t. the attack has an advantage of $\geq 10\%$. For comparison, we added the bit complexity $T_{\mathsf{pm}} = n^2 \cdot 2^{0.815 \cdot n}$ of the search algorithm of Dinur [Din21a].

Bounds for Macaulay Matrix-Based Algorithms and Implications for Multivariate Cryptography. We will see in Section 1.5 that there are natural connections between the algorithms presented here—which are based on algebraic relations—and Macaulay matrix-based algorithms, which are very popular in cryptanalysis [Fau99; CKPS00; Fau02; YC04; YC05; DBMMW08; MMDB08; TW10; Alb10; CCNY12]. This allows us to derive upper bounds for Macaulay matrix-based algorithms, as well.

Let $F : k^n \to k^m$ be a map of $m \geq n^{1+e}$ polynomials of degree d, and let $y \in k^m$. Let M_D be the Macaulay matrix for the equation system $F(X) = y$ up to degree D. Then, we have:

Informal Theorem 4. *1. There is a $D \in O(n^{1-e/(d-1)})$ s.t. M_D contains a contradiction with probability*

$$\geq 1 - O\left(\frac{n^{1-e/(d-1)}}{\#k}\right) \tag{22}$$

over the randomness of $y \leftarrow k^m$.

2. There is a $D \in O(\log(n)^{1/(d-1)} \cdot n^{1-e/(d-1)})$ s.t. M_D contains a contra-

diction with probability

$$\geq 1 - O\left(\frac{\log(n)^{1/(d-1)}}{n^{e/(d-1)}}\right) \tag{23}$$

over the randomness of $y \leftarrow k^m$.

For the multivariate search problem, we can prove the following upper bound in the average case:

Informal Theorem 5. *There is a $D \in O(\log(n)^{1/(d-1)} \cdot n^{1-e/(d-1)})$ s.t. for each $x \in k^n$ the Macaulay matrix M_D for the equation system $F(X) = F(x)$ contains the solutions $X_1 - x_1, \ldots, X_n - x_n$ with high probability over the randomness of F.*

This gives an upper bound for the hardness of uniformly random strongly overdetermined polynomial equation systems. It is left open how sharp these bounds are. Concretely, I ask the following:

Question 1. *Let $n, d \in \mathbb{N}$, $m > n$, and let k be a field of exponential size $\geq 2^n$. Let $D \in \mathbb{N}$ s.t.*

$$\binom{m+D}{D} \leq \binom{n+dD}{dD}. \tag{24}$$

Let $F : k^n \to k^m$ be a uniformly random polynomial map of degree d, and let $y \leftarrow k^m$ be uniformly random, too.

Is the probability that the Macaulay matrix M_{dD} of $F(X) = y$ up to degree $d \cdot D$ contains a contradiction negligible?

A positive answer to the above question would give us strong lower bounds for the performance of Macaulay matrix-based algorithms on solving uniformly random polynomial equation systems. Having such lower bounds is of emerging interest, since multivariate cryptosystems seem to be a viable alternative to lattice-based cryptosystems in a post-quantum world. An advantage of multivariate cryptosystems are that the only admissible attack vector against them seem to be attacks of algebraic nature. Hence, a positive answer to Question 1 would—under the hypothesis that there are only algebraic solvers for multivariate search problems—allow us to give precise estimates on the security of multivariate cryptosystems.

Lattice-Based Functional Encryption

In Chapter 2, we will revisit the results in [Üna20] and of my joint work with Erkan Tairi [TÜ23]. In fact, we will give completely new proofs for some results in both works and try to embed both works into one consistent framework.

Our results will consist of attacks on functional encryption schemes that adhere to the lattice-based FE framework we introduced in Informal Definition 1. Concretely, we will show the following:

Informal Theorem 6. *Let* FE *be a lattice-based functional encryption scheme in the sense of Informal Definition 1.*

1. *If* FE *is function-hiding, then it is not IND-CPA secure. Precisely, there is an adversary on the selective function-hiding IND-CPA security of* FE *that has a time complexity of* $\mathsf{poly}(q/p+\lambda)$ *and a non-negligible advantage.*
2. *Assume that* FE *supports the functionality of polynomials of degree* $d > 1$ *over* n *variables.*

   If the following requirements are met

   1. FE *is* linearly *compact, i.e., each ciphertext is a vector in* $\mathbb{Z}_q^m$ *where*

   $$m \in O(n), \tag{25}$$

   2. *and each secret key of* FE *is a polynomial in* $\mathbb{Z}_q[C_1, \ldots, C_m]$ *of degree* d*,*

   then FE *is not IND-CPA secure. Concretely, there is an adversary on the selective IND-CPA security of* FE *with a non-negligible advantage and time complexity in* $\mathsf{poly}(q/p+\lambda)$*.*

Unfortunately, the lower bound on compact lattice-based FE schemes is not as general, as we would like it to be. Concretely, we suspect:

Conjecture 1. *Let* FE *be a lattice-based FE scheme for polynomials of degree* $d > 1$ *over* n *variables. If ciphertexts of* FE *are compact, i.e., if there is an* $e > 0$ *s.t.*

$$m \in O(n^{d-e}), \tag{26}$$

then there is an IND-CPA adversary on FE *whose time-complexity and advantage are subexponential in* m*. I.e., there is a* $\delta < 1$ *s.t. the time complexity of the adversary lies in* $\mathsf{poly}(\lambda^{m^\delta})$*, while its advantage does not lie in* $\mathsf{negl}(\lambda^{m^\delta})$*.*

As we will see later, the hardship in proving the more general conjecture on compact lattice-based FE schemes will stem from a non-constructive argument that we will use in Section 2.5.3 to prove the algebraic homogeneity of ciphertexts. Concretely, we will be confronted with an increasing sequence of functions

$$0 = e_0(\lambda) \leq e_1(\lambda) \leq \ldots \leq e_{D-1}(\lambda) \leq e_D(\lambda) = 1. \tag{27}$$

If D is constant, then it is easy to see that we can draw a boundary between negligible and non-negligible functions, i.e., there must exist an $i \in \{0, \ldots, D-1\}$ s.t. $e_i(\lambda)$ is negligible, while $e_{i+1}(\lambda)$ is non-negligible. However, in the case of a non-constant D, we will fail to find such an index, since there is no algorithmic procedure to describe i. We will investigate this problem in greater detail in Section 2.5.3.

The adversaries that we give to prove Informal Theorem 6 will not rely on special lattice-based insights. In fact, our adversaries will use ad-hoc statistical and algebraic techniques and observations. However, because of conceptual simplicity, we will not directly attack lattice-based FE schemes. Instead, we will attack the simpler primitive of secret-key encryption over integers of constant depth, and reduce the security of lattice-based FE schemes over multiple cryptographic reductions and statistical hybrids to the security of such secret-key encryption schemes over the integers. We will explain this in the technical overview. We will also explain there how the lower bounds given here for function-hiding and compact FE can be circumvented.

Not Included Contributions

Not included in this thesis are two other works in which I participated:

Lower Bounds for Pairing-Based Verifiable Random Functions. In a joint work with Nicholas Brandt, Dennis Hofheinz and Julia Kastner [BHKÜ22], we showed lower bounds for the security of pairing-based verifiable random functions. A verifiable random function (VRF) [MRV99] is a pseudorandom function where the secret key-holder commits to a verification key and can issue proofs for output evaluations of the random function. The proofs enjoy the strong notion of unique provability, i.e., even when the verification key is malformed, it is impossible to prove that a certain input can evaluate under this verification key to two different outputs. While VRFs are high in demand, there are mostly pairing-based candidates known, which all have proofs of non-constant size or rely on assumptions of non-constant size [MRV99; Lys02; Dod03; DY05; ACF09; HW10; BMR10; KNP12; LLC15; Jag15; HJ16; Yam17; Bit17; GHKW17; Ros18; Koh19; Nie21].

To show our lower bounds, we introduced the notion of consecutively verifiable random functions and showed that each such VRF must express the exponents of its outputs as rational functions over the secret key elements. For such rational VRFs, we were then able to show the non-existence of algebraic black-box reductions that turn any VRF adversary into a solver of an Uber assumption. However, the class of the VRFs for which we could exclude the existence of such reductions is limited[8]. To prove our results, we needed to make use of ideal theory and algebraic elimination and extension theory to algebraically restrict the geometric set of explanations a reduction could give out for its output group elements. By this, we could show that an algebraic reduction cannot partition off one challenge point from other points queried by an adversary.

Compact Lattice-Based Selective Opening Secure Public-Key Encryption. In another joint work with Dennis Hofheinz, Kristina Hostáková, Julia Kastner and Karen Klein [HHKKÜ23], we constructed the first selective opening IND-CCA secure lattice-based public-key encryption scheme of constant ciphertext expansion whose security can be proven under LWE. A selective opening secure public-key encryption scheme (so-PKE) [DNRS99; BHY09] is a public-key encryption scheme which enjoys a strong security notion that guarantees that an adversary that may corrupt a subset of multiple senders and learn their messages and encryption randomness can not distinguish two possible sets of messages for unopened ciphertexts, *even* if the messages of opened and unopened ciphertexts are correlated. It is well known that normal IND-CCA security does not imply IND-CCA secure so-PKE [BDWY12; HR14; HRW16], and it turns out to be quite complicated to prove the selective opening security of traditional PKE schemes. Existing constructions [Hof12; BL17; LSSS17] are either based on non-post-quantum assumptions or inhibit a non-constant ciphertext expansion, i.e., the bit size of ciphertexts is not linear in the bit size of messages. To achieve compact so-PKE, we introduced compact lossy trapdoor functions

[8] In unpublished notes, I managed to broaden this class to include all consecutively verifiable random functions that base their security on Uber assumptions of constant size and have proofs and public keys consisting of a constant number of group elements.

and compact all-but-many lossy trapdoor functions, which we build upon the fully homomorphic encryption scheme dual-GSW [GSW13] mixed together with a packing technique for large numbers. Equipped with both lossy functions, we could follow the strategy of [HO13; Hof12] to construct generically a so-PKE. On the way, we fixed a flaw in this proof strategy.

Technical Overview

Algebraic Relations

For now, let k be any field, and denote by $k[X] = k[X_1, \dots, X_n]$ the corresponding polynomial ring over n variables.

Assume we are given $m \in \omega(n)$ polynomials $f_1, \dots, f_m \in k[X]$ of constant degree d. We can imagine that the polynomials $f_1, \dots, f_m$ are a—publicly known—part of a cryptosystem we want to analyse. Our aim is to understand the interplay of the polynomials $f_1, \dots, f_m$. For this end, let us bundle them to one polynomial map

$$\begin{aligned} F : k^n &\longrightarrow k^m && (28) \\ x &\longmapsto (f_1(x), \dots, f_m(x)) && (29) \end{aligned}$$

of degree d. Since $m > n$, the map F must have interesting properties. For example, the geometrical dimension of its image can at most be n, hence, it must lie thin in k^m. However, since the degree d of F might be two or larger, the image of F lies quite convoluted in k^m.

To understand the geometry of F, we will look at its dual morphism, which is a typical approach in algebraic geometry. Let $k[Y] = k[Y_1, \dots, Y_m]$ be the polynomial ring in m variables. The **dual morphism** of F is given by

$$\begin{aligned} \phi : k[Y] &\longrightarrow k[X] && (30) \\ Y_i &\longmapsto f_i(X). && (31) \end{aligned}$$

I.e., on input $h \in k[Y]$, ϕ substitutes each occurrence of Y_i in h by $f_i(X)$ for each $i \in [m]$. In other words, ϕ maps $h(Y_1, \dots, Y_m)$ to $h(f_1(X), \dots, f_m(X)) = h \circ F(X)$. Now, the algebraic relations of $f_1, \dots, f_m$ are exactly the kernel elements of ϕ. Since we assumed that $m > n$, the ring dimension m of $k[Y]$ is strictly larger than the ring dimension n of $k[X]$. Hence, ϕ must contain kernel elements. Indeed, we even know that $\ker \phi$ must be a prime ideal of height $m - n$. It follows that the existence of algebraic relationships of $f_1, \dots, f_m$ is guaranteed for $m > n$. However, the degree of algebraic relations $h \in \ker \phi$ might be exponentially large. As usual, high-level algebra yields the qualitative result (the existence of algebraic relations), however, for the quantitative result (bounding the degree of algebraic relations), we need to delve a bit deeper into the details of ϕ. For this end, note that ϕ has a nice structure, since it is a ring morphism. Additionally, since each f_i is of degree d, we can control the expansion of ϕ in $k[X]$. To make this precise note that we have for each $L \in \mathbb{N}$

$$\phi\big(k[X]^{\leq L}\big) \subseteq k[Y]^{\leq dL}. \qquad (32)$$

I.e., whenever $h \in k[Y]$ is of total degree $\leq L$, the degree of $\phi(h) = h(f_1, \dots, f_m)$ must be bounded by $\leq dL$. Hence, let ϕ^L be the restriction of the ring morphism

ϕ to the subspace $k[Y]^{\leq L}$ of polynomials of degree L

$$\phi^L : k[Y]^{\leq L} \longrightarrow k[X]^{\leq dL}. \tag{33}$$

ϕ^L is not a ring morphism, however it is still linear. Of importance for us is that ϕ^L contains exactly the algebraic relations of degree $\leq L$. Hence, it suffices to find a minimum $L \in \mathbb{N}$ s.t. we can prove $\ker \phi^L \neq 0$. To show this we can apply the dimension formula, which yields

$$\dim_k \ker \phi^L \geq \dim_k k[Y]^{\leq L} - \dim_k k[X]^{\leq dL}. \tag{34}$$

The dimensions $\dim_k k[Y]^{\leq L} = \binom{m+L}{L}$ and $\dim_k k[X]^{\leq dL} = \binom{n+dL}{dL}$ are well known. Hence, it suffices to find an $L \in \mathbb{N}$ s.t.

$$\dim_k \ker \phi^L \geq \binom{m+L}{L} - \binom{n+dL}{dL} > 0. \tag{35}$$

We claim that there is an $L \in O\big((n^d/m)^{1/(d-1)}\big)$ that fulfils the inequality $\binom{m+L}{L} > \binom{n+dL}{dL}$. Let us motivate why such an L exist: assume for simplicity that n is larger than dL (this will hold for n large enough). Now, $\binom{m+L}{L}$ can be lower bounded by

$$\binom{m+L}{L} \geq \left(\frac{m+L}{L}\right)^L \geq \frac{m^L}{L^L}. \tag{36}$$

On the other hand, $\binom{n+dL}{dL}$ can be upper bounded by

$$\binom{n+dL}{dL} \leq \binom{2n}{dL} \leq e^{dL} \cdot \left(\frac{2n}{dL}\right)^{dL} = (2e)^{dL} \cdot \frac{n^{dL}}{(dL)^{dL}}, \tag{37}$$

where e denotes Euler's number. Hence, Eq. (35) is implied by the following stricter inequality

$$\frac{m^L}{L^L} > (2e)^{dL} \cdot \frac{n^{dL}}{(dL)^{dL}}. \tag{38}$$

We can now easily take the L-th root and move $(dL)^d$ to the left side. This yields the equivalent inequality

$$d^d \cdot m \cdot L^{d-1} > (2e)^d \cdot n^d. \tag{39}$$

By solving for L, we get

$$L > \left(\frac{2e}{d}\right)^d \cdot \left(\frac{n^d}{m}\right)^{1/(d-1)} \in O\left(\left(\frac{n^d}{m}\right)^{1/(d-1)}\right). \tag{40}$$

Hence, for a sufficiently large $L \in O\Big(\big(n^d/m\big)^{1/(d-1)}\Big)$, Eq. (35) is fulfilled and an algebraic relationship $h \in \ker \phi$ of degree $\leq L$ must exist.

Note that L only depends on n, m, d and is independent of the concrete map $F : k^n \to k^m$ and of the field k. It follows that for each function $m \in \omega(n)$ and each constant d, there exists a universal function $D \in O\Big(\big(n^d/m\big)^{1/(d-1)}\Big)$, s.t., for each possible ensemble $f_1, \dots, f_m \in k[X]$ of polynomials of degree d, there exists a polynomial $h \in k[Y]$ that meets the following requirements:

$$h \neq 0 \in k[Y], \tag{41}$$
$$h(f_1, \dots, f_m) = 0 \in k[X], \tag{42}$$
$$\deg h \leq D(n). \tag{43}$$

Now, from an algorithmic point of view, we need to clarify how h can be computed when given $f_1, \dots, f_m$. Luckily, the answer for this question follows directly from the proof of existence of h: it suffices to consider the restriction of the dual morphism

$$\phi^D : k[Y]^{\leq D} \longrightarrow k[X]^{\leq dD}. \tag{44}$$

ϕ^D is a linear map of two well-known vector spaces. Given an input $h \in k[Y]^{\leq D}$, it is easy to evaluate $\phi^D(h) = h(f_1(X), \dots, f_m(X))$ as long as the polynomials $f_1, \dots, f_m$ are known. Hence, we can directly compute a matrix representation $M \in k^{\binom{n+dD}{dD} \times \binom{m+D}{D}}$ of ϕ^D. Finding kernel elements of ϕ^D now corresponds to finding kernel vectors of M, which can be done by Gaussian elimination. Since $m \in \omega(n)$, $D \in O\Big(\big(n^d/m\big)^{1/(d-1)}\Big)$ is sublinear. The arithmetic cost for finding h is therefore bounded by the subexponential function class

$$O\left(\binom{m+D}{D} \cdot \binom{n+dD}{dD}^2\right) \subseteq n^{O\left(\left(n^d/m\right)^{1/(d-1)}\right)}. \tag{45}$$

This should answer our questions about the existence of algebraic relationships h of polynomials $f_1, \dots, f_m$, bounds for their degree and algorithmic ways to compute them. However, before we turn to the applications of h, let us get back to the geometrical interpretation of the map F, with which we started. What can h tell us about the geometry of the image of F?

We know that the image of F is contained in the set $V(h)$ of roots of h. The set $V(h)$ is a hypersurface of codimension 1, i.e., it lies very thin in k^m. Unfortunately, the degree of h may be non-constant, hence the $V(h)$ might lie very complicated in k^m. However, if the degree of h is minimal among non-zero elements of ϕ, then we can see $V(h)$ as a first approximation that separates the image of F from k^m. In general, if $h_1, \dots, h_\ell \in k[Y]$ generate the ideal $\ker \phi$, then Hilbert's Nullstellensatz implies that the set $V(h_1, \dots, h_\ell)$ of common roots of $h_1, \dots, h_\ell$ is exactly the Zariski closure of the image of F. This means that $V(h_1, \dots, h_\ell)$ is the minimal algebraic set that contains $\operatorname{Img} F$. Now, $\operatorname{Img} F$ might not be an algebraic set itself, however it lies dense in $V(h_1, \dots, h_\ell)$. I.e., up to some points, $\operatorname{Img} F$ is fully described by the generators $h_1, \dots, h_\ell$ of algebraic relations of $f_1, \dots, f_m$.

Relations over Small Fields

A careful observer might have noticed that the case where k is a small field might raise problems. For example, if $k = \mathbb{Z}_2$, it might be that we compute the polynomial $h(Y) = Y_1^2 - Y_1$ as an algebraic relationship of ϕ. Now, h will later not turn out to be useful, since it vanishes on the whole space $\mathbb{Z}_2^m$ and does not separate the image of F from $\mathbb{Z}_2^m$.

The core of this problem stems from the fact that the rings $k[Y]$ and $k[X]$ are not the algebraically correct coordinate rings of k^m and k^n. To solve this problem, we can divide out the field equations from the polynomial rings $k[Y]$ and $k[X]$. Let $q = \#k$ be the size of k, then the reduced dual morphism

$$\begin{aligned} \phi : k[Y]/(Y_1^q - Y_1, \dots, Y_m^q - Y_m) &\longrightarrow k[X]/(X_1^q - X_1, \dots, X_n^q - X_n) \\ Y_i &\longmapsto f_i(X) \end{aligned}$$

is well-defined. Now, by applying the same rationale as above, we can show that $\ker \phi$ contains an element h with degree in $O\Big(\big(n^d/m\big)^{1/(d-1)}\Big)$ that does not lie in $(Y_1^q - Y_1, \dots, Y_m^q - Y_m)$. We will deal with this procedure in larger detail in Section 1.2.1.

On Algebraic Pseudorandom Generators

We can directly use the existence of algebraic relations to attack algebraic PRGs. For this end, let

$$F : k^n \longrightarrow k^m \tag{46}$$

be a PRG that maps n random elements of a finite field k to m pseudorandom elements of k. Assume that F is algebraic, in the sense that there are m polynomials $f_1, \dots, f_m \in k[X]$ of constant degree d that compute the output values of F. If $m \in \omega(n)$, then we know that there must be an algebraic relation h among $f_1, \dots, f_m$ of degree $D \in O\Big(\big(n^d/m\big)^{1/(d-1)}\Big)$. As we explained above, h separates the image of F from the whole space k^m. This gives rise to the following adversary $\mathcal{A}$: given a point $y \in k^m$, $\mathcal{A}$ evaluates h on y. If $h(y) = 0$, then $\mathcal{A}$ decides that y lies in the image of F and outputs 0. Otherwise, $\mathcal{A}$ decides that y has been sampled uniformly at random from k^m and outputs 1.

Now, if $y = F(x)$ for $x \leftarrow k^n$, then $\mathcal{A}$ will always output 0. This is, because h must vanish on the image of F. On the other hand, if $y \leftarrow k^m$ is sampled uniformly at random, the probability that the non-zero polynomial h does not vanish on k is according to the Schwartz-Zippel lemma lower bounded by

$$\Pr_{y \leftarrow k^m}[h(y) \neq 0] \geq 1 - \frac{\deg h}{\#k} \in 1 - \frac{1}{\#k} \cdot O\Big(\big(n^d/m\big)^{1/(d-1)}\Big) \subseteq 1 - \frac{1}{\#k} \cdot o(n).$$

Hence, the advantage of $\mathcal{A}$ is at least $1 - o(n/\#k)$. If k is large enough, i.e. $\#k \geq n$, $\mathcal{A}$ has a high advantage.

The time complexity of $\mathcal{A}$ is dominated by computing h when given F and evaluating h at y. Computing h costs $n^{O(D)}$ arithmetic operations over k. Evaluating h also costs $D \cdot \binom{m+D}{D} \in n^{O(D)}$ arithmetic operations. This yields a subexponential time complexity of $\mathcal{A}$ when $m \in \omega(n)$ is superlinear. In particular, if $m \geq n^{1+e}$ for a constant $e > 0$, then the degree of h lies in $D \in O(n^{1-e/(d-1)})$ and the time complexity of $\mathcal{A}$ is bounded by $n^{O(n^{1-e/(d-1)})}$.

Using Field Extensions over Small Fields

The above attack works well if the size of k is larger than n, however, it may fail for small fields. A direct solution for this problem would be to use an algebraic relationship h that is reduced modulo the field equations of k. This would retain the time complexity of $\mathcal{A}$. However, for a reduced relationship h, we would only be able to prove a non-trivial, but subexponentially small advantage of $\mathcal{A}$.

Instead, we will present here an elegant solution by using field extensions. Concretely, let $k \subset \overline{k}$ be a field extension of degree $r = [\overline{k} : k] = \dim_k \overline{k}$. We assume the degree r of the extension has been chosen minimal with $\#\overline{k} = \#k^r \geq n$. Hence, $r \leq \lceil \log(n) \rceil$. Let ζ be a generator of the extension $\overline{k} = k[\zeta]$. Then, each element $a \in \overline{k}$ can be written as

$$a = b_1 + b_2 \cdot \zeta + \ldots + b_r \cdot \zeta^{r-1} \tag{47}$$

with $b_1, \ldots, b_r \in k$. In particular, the k-linear map

$$\psi : k^r \longrightarrow \overline{k} \tag{48}$$
$$(b_1, \ldots, b_r) \longmapsto b_1 + b_2 \cdot \zeta + \ldots + b_r \cdot \zeta^{r-1} \tag{49}$$

is an isomorphism of k-vector spaces.

Now, let $F : k^n \to k^m$ be a PRG of degree d over k. Let us for simplicity assume that r divides m, and let us set $m' = m/r$. Our idea is to turn F into a PRG $G : k^n \to \overline{k}^{m'}$, whose values are computed by polynomials

$$g_j := f_{1+r\cdot(j-1)} + f_{2+r\cdot(j-1)} \cdot \zeta + \ldots + f_{r\cdot j} \cdot \zeta^{r-1} \tag{50}$$

for $j \in [m']$. By abuse of notation, we can also write

$$G := \psi \circ F \tag{51}$$

where ψ gets applied block-wise on the polynomials $f_1, \ldots, f_m$. While the $f_1, \ldots, f_m$ are polynomials of degree d over the base field k, the $g_1, \ldots, g_{m'}$ are polynomials of the same degree d over the extension field $\overline{k}$.

We claim that G is a PRG iff F is one. Concretely, we mean that distinguishing an image point $G(x)$, $x \leftarrow k^n$, from a truly random point $y' \leftarrow \overline{k}^{m'}$ is equivalent to distinguishing an image $F(x)$, $x \leftarrow k^n$, from a truly random point $y \leftarrow k^m$. Indeed, let $\mathcal{A}$ be an adversary on the pseudorandomness of G, and let us give a reduction $\mathcal{R}$ that attacks the pseudorandomness of F. When given $y \in k^m$, our strategy is to compute $y' := \psi(y) \in \overline{k}^{m'}$, i.e.

$$y'_j := y_{1+r\cdot(j-1)} + y_{2+r\cdot(j-1)} \cdot \zeta + \ldots + y_{r\cdot j} \cdot \zeta^{r-1}, \tag{52}$$

to submit (G, y') to $\mathcal{A}$ and to pass on the output of $\mathcal{A}$. We claim that the advantage of this reduction equals the advantage of $\mathcal{A}$ against G. Indeed, if $y = F(x)$ is an image point of F, then

$$y' = \psi(y) = \psi(F(x)) = (\psi \circ F)(x) = G(x) \tag{53}$$

is an image point of G. On the other hand, if $y \leftarrow k^m$ is uniformly random, then $y' = \psi(y)$ is also uniformly distributed in $\overline{k}^{m'}$. This is, because by applying ψ blockwise we get an isomorphism

$$\psi : k^m \to \overline{k}^{m'}. \tag{54}$$

Hence, the view of $\mathcal{A}$ when interacting with $\mathcal{R}$ is identical to $\mathcal{A}$'s view when attacking the pseudorandomness of G.

Now, let $\mathcal{A}$ be the adversary that on input (G, y') computes an algebraic relationship $h \in \overline{k}[Y_1, \ldots, Y_{m'}]$ of the polynomials $g_1, \ldots, g_{m'}$ and outputs 0 iff $h(y') = 0$. We can bound the degree of h by

$$\deg h \in O\Big((n^d/m')^{1/(d-1)}\Big) \subseteq O\Big((\log(n) \cdot n^d/m)^{1/(d-1)}\Big) \tag{55}$$

where we used $m' = m/r \le m/\lceil \log n \rceil$. For $m \in \omega(\log(n) \cdot n)$, we now get an advantage for $\mathcal{A}$ of

$$\ge 1 - \frac{\deg h}{\#\overline{k}} \ge 1 - \frac{1}{\#\overline{k}} O\Big((\log(n) \cdot n^d/m)^{1/(d-1)}\Big) \ge 1 - \frac{1}{n} \cdot o(n) \ge 1 - o(1).$$

Further, the time complexity of $\mathcal{A}$ lies in $n^{O\left((\log(n) \cdot n^d/m)^{1/(d-1)}\right)}$. The success probabilities for $\mathcal{A}$ when attacking $G : k^n \to \overline{k}^{m'}$ directly transfer to success probabilities for $\mathcal{R}$ when attacking $F : k^n \to k^m$. It follows, that $\mathcal{R}$ has a high advantage of distinguishing image points of F from uniformly random points, while making use of $n^{O\left((\log(n) \cdot n^d/m)^{1/(d-1)}\right)}$ arithmetic operations over $\overline{k}$.

Summarizing, for a large enough stretch $m \in \omega(\log(n) \cdot n)$, we can attack the pseudorandomness of every algebraic PRG $F : k^n \to k^m$ over any field k in subexponential time with high advantage. The concrete time complexity depends on the degree d of F and the stretch m, and improves with larger m and smaller d. As a next step, one can use the distinguishing attacks here to invert the map F in the average case. Further, it is possible to establish connections between the attacks presented here and classical Macaulay matrix-based attacks, which are quite popular in algebraic cryptanalysis. However, we will postpone the detailed discussion of both things to the technical work in Sections 1.4 and 1.5 and continue with applying algebraic relations in the field of lattice-based functional encryption.

On Lattice-Based Functional Encryption

Let us now turn to our investigation of lattice-based FE schemes. Unfortunately, we cannot apply algebraic relations directly here. Indeed, we will first have to turn the lattice-based nature of the studied FE schemes into an algebraic structure, which we can then analyse by algebraic methods. For this end, we will have to introduce all tools that were developed in [Üna20].

I decided to do this in a bottom-up fashion. I.e., we will start with simple problems, which are seemingly uncorrelated to our objective of study, and subsequently derive solutions for more complex problems, until we arrive at lattice-based FE schemes. Concretely, we will first study the problem of distinguishing different distributions by the mean of their squares, then the IND-CPA security of simple SKE schemes of constant depth and then arrive at lattice-based FE.

On Mean Square Distinguishers

Let $\mathcal{D}_0, \mathcal{D}_1$ be two memoryless discrete distributions over $\mathbb{Z}$ and consider the problem of distinguishing $\mathcal{D}_0$ from $\mathcal{D}_1$. Concretely, when given samples

$$\alpha_1, \ldots, \alpha_N \leftarrow \mathcal{D}_0, \qquad \beta_1, \ldots, \beta_N \leftarrow \mathcal{D}_1 \qquad \text{and} \qquad \gamma_1, \ldots, \gamma_N \leftarrow \mathcal{D}_b, \tag{56}$$

we are expected to determine the uniformly random bit $b \leftarrow \{0,1\}$. For the above problem, there might be a lot of different solutions and algorithms. We will study here a very simple approach at distinguishing, for which we are able to give formal guarantees for the following kind of special distributions: let $\mathcal{E}$ be a discrete memoryless distribution of univariate polynomials in $\mathbb{Z}[X]$ of constant degree d. For $x \in \{0, \dots, 2d\}$, let $\mathcal{D}_x$ be the distribution that is obtained by sampling $f \leftarrow \mathcal{E}$ and outputting $f(x)$. Assume that the output of each $\mathcal{D}_x$ is bounded by some $B > 0$, and assume that the probability

$$\Pr_{f \leftarrow \mathcal{E}}[\deg f > 0] \tag{57}$$

is non-negligible. We ask if there are $x, y \in \{0, \dots, 2d\}$ s.t. $\mathcal{D}_x$ and $\mathcal{D}_y$ can be distinguished with non-negligible advantage when only given $N \in \mathsf{poly}(B + \lambda)$ samples.

The answer turns out to be yes. Let us first sketch a simplified version of the distinguishing algorithm $\mathcal{T}$ that we will use for this task. On input the distributions $\mathcal{D}_0, \mathcal{D}_1$ and $\mathcal{D}_b$, the algorithm $\mathcal{T}(\mathcal{D}_0, \mathcal{D}_1, \mathcal{D}_b)$ will proceed as follows:

```
draw $\alpha_1, \dots, \alpha_N \leftarrow \mathcal{D}_0$
draw $\beta_1, \dots, \beta_N \leftarrow \mathcal{D}_1$
draw $\gamma_1, \dots, \gamma_N \leftarrow \mathcal{D}_b$
set $\overline{\alpha} := \frac{1}{N} \sum_{i=1}^{N} \alpha_i^2$
set $\overline{\beta} := \frac{1}{N} \sum_{i=1}^{N} \beta_i^2$
set $\overline{\gamma} := \frac{1}{N} \sum_{i=1}^{N} \gamma_i^2$
set $e_0 := |\overline{\alpha} - \overline{\gamma}|$
set $e_1 := |\overline{\beta} - \overline{\gamma}|$
if $e_0 > e_1$ then
    return $1$
else if $e_1 > e_0$ then
    return $0$
else
    draw $b' \leftarrow \{0,1\}$
    return $b'$
end if
```

In other words, $\mathcal{T}$ approximates the values

$$\overline{\alpha} \approx \mathop{\mathbb{E}}_{\alpha \leftarrow \mathcal{D}_0}[\alpha^2], \qquad \overline{\beta} \approx \mathop{\mathbb{E}}_{\beta \leftarrow \mathcal{D}_1}[\beta^2] \qquad \text{and} \qquad \overline{\gamma} \approx \mathop{\mathbb{E}}_{\gamma \leftarrow \mathcal{D}_b}[\gamma^2]. \tag{58}$$

If $\overline{\gamma}$ is closer to $\overline{\alpha}$ than to $\overline{\beta}$, then $\mathcal{T}$ rules that $\overline{\alpha}$ and $\overline{\gamma}$ must be equally distributed. Otherwise, if the distance between $\overline{\beta}$ and $\overline{\gamma}$ is smaller, then $\mathcal{T}$ decides that the γ values stem from $\mathcal{D}_1$. If the distributions $\mathcal{D}_0, \mathcal{D}_1$ are bounded by B, then by setting $N \in \mathsf{poly}(\lambda + B)$ large enough, $\mathcal{T}$ can approximate the means $\mathbb{E}[\alpha^2]$, $\mathbb{E}[\beta^2]$ and $\mathbb{E}[\gamma^2]$ up to arbitrarily small polynomial fractions. Hence, we can guarantee a high distinguishing advantage of $\mathcal{T}$ whenever the distance between $\mathbb{E}[\alpha^2]$ and $\mathbb{E}[\beta^2]$ is non-negligible.

To come back to our original problem, remember that the distributions $\mathcal{D}_x$, $x \in \{0, \dots, 2d\}$, are given by sampling a univariate degree-d polynomial $f \leftarrow \mathcal{E}$ and outputting $f(x)$. In Section 2.2.3, we will show that, if the distances

$$\left| \mathop{\mathbb{E}}_{\alpha \leftarrow \mathcal{D}_x}[\alpha^2] - \mathop{\mathbb{E}}_{\beta \leftarrow \mathcal{D}_y}[\beta^2] \right| = \left| \mathop{\mathbb{E}}_{f \leftarrow \mathcal{E}}[f(x)^2] - \mathop{\mathbb{E}}_{f \leftarrow \mathcal{E}}[f(y)^2] \right| \tag{59}$$

are negligible for all $x, y \in \{0, \dots, 2d\}$, then the random polynomial $f \leftarrow \mathcal{E}$ will be constant with overwhelming probability (and, hence, the statistical distance between $\mathcal{D}_x, \mathcal{D}_y$ will be negligible, anyway). Let us close the discussion on mean square distinguishing by sketching a proof for this claim. Our proof strategy is based on induction on $d \in \mathbb{N}$:

1. The case $d = 1$:
   Let $\mathcal{E}$ be a distribution of univariate degree-1 polynomials $f = aX + b \in \mathbb{R}[X]$, and assume that we have for all $x, y \in \{0, 1, 2\}$

   $$\left| \mathop{\mathbb{E}}_{f \leftarrow \mathcal{E}}[f(x)^2] - \mathop{\mathbb{E}}_{f \leftarrow \mathcal{E}}[f(y)^2] \right| \in \mathsf{negl}(\lambda). \tag{60}$$

   We claim that almost always f must be constant, i.e., a must vanish with overwhelming probability. Indeed, by setting $x = 1, y = 0$ in Eq. (60), we get

   $$\left|\mathbb{E}[f(1)^2] - \mathbb{E}[f(0)^2]\right| = \left|\mathbb{E}[(a+b)^2] - \mathbb{E}[b^2]\right| = \left|\mathbb{E}[a^2] + 2\,\mathbb{E}[ab]\right| \in \mathsf{negl}(\lambda).$$

   For $x = 2$ and $y = 0$, we can deduce

   $$\left|\mathbb{E}[f(2)^2] - \mathbb{E}[f(0)^2]\right| = \left|\mathbb{E}[(2a+b)^2] - \mathbb{E}[b^2]\right| = \left|4\,\mathbb{E}[a^2] + 4\,\mathbb{E}[ab]\right| \in \mathsf{negl}(\lambda).$$

   Now, the trick is to use the reverse triangle inequality $|u| \leq |v| + |v - u|$ to show

   $$\left|\mathbb{E}[a^2]\right| \leq \left|2\,\mathbb{E}[a^2] + 2\,\mathbb{E}[ab]\right| + \left|\mathbb{E}[a^2] + 2\,\mathbb{E}[ab]\right| \in \mathsf{negl}(\lambda). \tag{61}$$

   At this point, we need to remember that f and its coefficients are integer. Hence, we can bound the probability of a being non-zero by the mean of its square as follows:

   $$\Pr[a \neq 0] = \sum_{z \in \mathbb{Z} \setminus \{0\}} \Pr[a = z] \leq \sum_{z \in \mathbb{Z} \setminus \{0\}} \Pr[a = z] \cdot z^2 = \mathbb{E}[a^2] \in \mathsf{negl}(\lambda).$$

   Now, a is the leading coefficient of $f = aX + b$. If a vanishes with overwhelming probability, then f must be constant in an overwhelming number of cases, as we claimed.

2. The case $d > 1$:
   Now, $f = c_d X^d + \ldots + c_1 X + x_0$ consists of $d+1$ random (and potentially correlated) coefficients. Our idea is to prove that the leading coefficient c_d is zero with overwhelming probability. Afterwards, we can assume that f is always of degree $\leq d-1$ (this will only introduce a statistically negligible error) and invoke the induction hypothesis for $\deg f < d$.

   We will again bound $\mathbb{E}[c_d^2]$ by invoking Eq. (60) on pairs $x = 0, \ldots, 2d$, $y = 0$ and linearly combining the resulting inequalities.

   First we note that, by using discrete derivations, it is possible to show that for any polynomial $g(X) = b_\ell \cdot X^\ell + \ldots + b_1 \cdot X + b_0$ it holds

   $$\ell! \cdot b_\ell = \sum_{i=0}^{\ell} (-1)^{\ell - i} \binom{\ell}{i} \cdot b(i). \tag{62}$$

   We can apply this equality to the polynomial $g(X) = \mathbb{E}[f(X)^2] - \mathbb{E}[f(0)^2]$, whose leading term is $\mathbb{E}[a_d^2]$. This yields

   $$(2d)! \cdot \mathbb{E}[a_d^2] = \sum_{i=0}^{2d} (-1)^{2d-i} \binom{2d}{i} \cdot \left(\mathbb{E}[f(i)^2] - \mathbb{E}[f(0)^2]\right). \tag{63}$$

Now, by taking absolute values, we get

$$\begin{aligned}
&(2d)! \cdot \left|\mathbb{E}[a_d^2]\right| && (64)\\
=&\left|\sum_{i=0}^{2d}(-1)^{2d-i}\binom{2d}{i} \cdot \left(\mathbb{E}[f(i)^2] - \mathbb{E}[f(0)^2]\right)\right| && (65)\\
\leq&\sum_{i=0}^{2d}\binom{2d}{i} \cdot \left|\mathbb{E}[f(i)^2] - \mathbb{E}[f(0)^2]\right| \in \mathsf{negl}(\lambda), && (66)
\end{aligned}$$

where the sum in the last line must be negligible, since each distance $\left|\mathbb{E}[f(i)^2] - \mathbb{E}[f(0)^2]\right|$ is negligible. It follows that $\mathbb{E}[a_d^2]$ is negligible, too, and that a_d does almost always vanish.

Hence, by a statistical hybrid step, we can assume that $f \leftarrow \mathcal{E}$ is always of degree $\leq d-1$. By an induction argument, it follows that all coefficients $a_d, \ldots, a_1$ of f must almost always be zero.

SKE over the Integers

Let us put our mean square distinguisher $\mathcal{T}$ at use by analysing secret-key encryption schemes over the integers of constant depth. For this end, we will first have to introduce some notions regarding encryption algorithms Enc.

Informal Definition 2 (Encryption of Small Depth and Width)**.** Let R be a ring with a (quasi-)valuation $|\cdot| : R \to \mathbb{R}_{\geq 0}$. For example, $R = \mathbb{Z}$ with the Archimedean valuation, or $R = \mathbb{Z}_q$ with $|z \bmod q| := \min_{a \in q\mathbb{Z}+z} |a|$.

We will assume that on input a master secret key msk and an integer message $x \in \mathcal{X} \subset \mathbb{Z}^n$, the algorithm Enc always outputs a vector $\mathsf{ct} \in R^m$ as ciphertext.

1. We say that Enc is of **depth** d over R, if two things are satisfied:

   First, there is a so-called **offline algorithm** $\mathsf{Enc_{off}}$ that—on input msk—outputs m polynomials $r_1, \ldots, r_m \in R[X_1, \ldots, X_n]$ of degree d.

   Second, on input msk and $x \in \mathbb{Z}^n$, $\mathsf{Enc}(\mathsf{msk}, x)$ samples $(r_1, \ldots, r_m) \leftarrow \mathsf{Enc_{off}}(\mathsf{msk})$ and outputs

   $$\mathsf{ct} := (r_1(x), \ldots, r_m(x)) \in R^m \qquad (67)$$

   as ciphertext for x.

2. We say that Enc is of **width** $B > 0$ over R, if we have for each ciphertext $\mathsf{ct} = (c_1, \ldots, c_m)$ outputted by Enc

   $$||\mathsf{ct}||_\infty := \max_{j=1,\ldots,m} |c_j| \leq B. \qquad (68)$$

Being of constant depth implies that Enc can be separated into an offline part $\mathsf{Enc_{off}}$ and an online part. The offline part is allowed to be arbitrarily complex, however, it only sees the master secret key and is agnostic to the to-be-encrypted message. It produces correlated randomness, that is used by Enc in its online part to generate the final ciphertext. The crucial point is that this final step is very simple from an algebraic point of view and can be implemented by an arithmetic circuit of constant depth.

We note that offline/online separation is of its own interest, since it follows a design principle, where a device with limited power may generate encryption randomness a priori (when it has a lot of power), and then economically encrypts messages online, while it might not have a lot of power any more [IPS08; HW14; AAB15].

Most lattice-based encryption algorithms are of depth 1 over some ring $\mathbb{Z}_q$. Typically, these algorithms generate uniform random vectors and Gaussian noise values in the offline phase. In the online phase, they add a scaled version of the message vector to the random vector generated in the offline phase and output the sum as ciphertext. However, only considering encryption algorithms of degree 1 would not suffice for compact FE schemes for polynomials of higher degree. Therefore, it makes sense to allow for constant degrees larger than 1 in our definition. A typical lattice-based operation that is not covered by our definition is bit-decomposition. Bit-decomposition over large fields has a very high degree. If Enc decomposes the message x into its bits, or if the ciphertext ct gets decomposed into bits at some later points (which is very typical for fully homomorphic encryption schemes), then our lower bounds do not apply, any more. At the end of this subsection, we will discuss further limitations for our framework.

Finally, let us note that the last property, Enc being of bounded width is to ensure that our mean square distinguisher can approximate its means correctly.

Now, for an SKE of bounded width and constant depth over the integers, we can show the following:

Informal Theorem 7 (Theorem 77)**.** *Let* $\mathsf{SKE} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{Dec})$ *be a secret-key encryption scheme for messages* $\mathcal{X} = \{0, \ldots, 2d\}$ *s.t.* Enc *is of constant depth d and width B over $\mathbb{Z}$. Denote the* ***decryption probability*** *of* SKE *by*

$$\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}) := \min_{x \in \mathcal{X}} \left(\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)} [\mathsf{Dec}(\mathsf{msk}, \mathsf{Enc}(\mathsf{msk}, x)) = x] \right). \tag{69}$$

Then, if $\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}) - \frac{1}{\#\mathcal{X}}$ *is non-negligible, there is an adversary $\mathcal{A}$ that has a non-negligible advantage against the selective IND-CPA security of* SKE *and a time complexity of* $\mathsf{poly}(\lambda + B)$.

Let us explain why we require $\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec})$ to be larger than $1/(2d + 1)$ by a non-negligible amount. At encryption, $\mathsf{Enc}(\mathsf{msk}, x)$ samples $(r_1, \ldots, r_m) \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$ and outputs $\mathsf{ct} = (r_1(x), \ldots, r_m(x))$ as ciphertext. If the polynomials $r_1, \ldots, r_m$ are almost always constant, then the distribution of ct is with overwhelming probability independent of x. In particular, a meaningful decryption in this case would be impossible. However, Dec could still attempt to *decrypt* a ciphertext by outputting a uniformly random message $y \leftarrow \{0, \ldots, 2d\}$, which leads to a correct result with probability $1/(2d + 1)$. Hence, even if all ciphertexts are devoid of any information about their messages, $\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec})$ can still be $1/(2d + 1)$.

On the other hand, if $\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec})$ is larger than $1/(2d+1)$ by a non-negligible amount, then it is easy to show that there is a $j \in [m]$ s.t. with some non-negligible probability the polynomial r_j is not constant for $(r_1, \ldots, r_m) \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$. Now, for $x \in \{0, \ldots, 2d\}$, we can consider the distribution $\mathcal{D}_x$ that samples $(r_1, \ldots, r_m) \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$ and outputs $r_j(x)$. As we have seen before, there must be $x, y \in \{0, \ldots, 2d\}$, s.t. our mean square distinguisher $\mathcal{T}$

can distinguish between $\mathcal{D}_x$ and $\mathcal{D}_y$ with non-negligible advantage by evaluating $N \in \mathsf{poly}(\lambda + B)$ samples.

This leads to the following adversary $\mathcal{A}$:

Step 1: $\mathcal{A}$ plays the selective IND-CPA security game of SKE with a challenger $\mathcal{C}$. At start of the game, $\mathcal{A}$ draws $x, y \leftarrow \{0, \dots, 2d\}$ and $j \leftarrow [m]$ uniformly and independently at random.

Step 2: $\mathcal{A}$ asks $\mathcal{C}$ for N encryptions of x, N encryptions of y and queries N challenge ciphertexts for the candidate message pair (x, y).

Step 3: $\mathcal{A}$ receives $3N$ ciphertexts from $\mathcal{C}$ and forwards the j-th coordinate from each ciphertext to $\mathcal{T}$. Finally, it returns the output from $\mathcal{T}$ to the challenger.

Note, that the j-th coordinate of the first N ciphertexts is distributed according to $r_j(x)$, while the j-th coordinate of the following N ciphertexts is distributed according to $r_j(y)$. Since $\mathcal{A}$ submitted the candidate message pair (x, y), the j-th coordinate of the last N ciphertexts that $\mathcal{A}$ receives from $\mathcal{C}$ are either distributed according to $r_j(x)$, if the secret bit b of $\mathcal{C}$ is zero, or according to $r_j(y)$, if $b = 1$. Since r_j is not constant with non-negligible probability, $\mathcal{T}$ has a non-negligible advantage at guessing b correctly. Hence, the non-negligible advantage of $\mathcal{A}$ follows.

SKE of Small Width

Above, we attacked SKE schemes of constant depth over the integers. Since lattice-based schemes are given over rings[9] $\mathbb{Z}_q$, we will need an intermediate SKE scheme to build a bridge from our above attack to lattice-based schemes. This class of intermediate schemes will be SKE schemes of constant depth d and bounded width B over $\mathbb{Z}_q$. We will require that there is a large enough gap between B and q. A bit more formally, we will show:

Informal Theorem 8 (Theorem 75)**.** *Let q be prime, d constant and $B > 0$ s.t.*

$$B < q/C \tag{70}$$

for a constant C that only depends on d.

Let $\mathsf{SKE} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{Dec})$ be a secret-key encryption scheme of depth d and width B over $\mathbb{Z}_q$ with message space $\{0, \dots, 2d\}$. If the decryption probability $\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec})$ of Dec is by a non-negligible amount larger than $1/(2d+1)$, then there is an adversary $\mathcal{A}$ that performs $\mathsf{poly}(\lambda + B)$ arithmetic operations and has a non-negligible advantage in the selective IND-CPA security game of SKE.

Note that the requirement for the width in Eq. (70) is necessary. In fact, it is easy to construct SKEs over $\mathbb{Z}_q$ of depth 1 whose security can be reduced to the hardness of the LWE problem. In some sense, Informal Theorem 8 states that it is impossible to achieve secure encryption from LWE with (geometrically) small ciphertexts (at least without using bit decomposition).

An idea to prove Informal Theorem 8 is to construct an SKE $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \mathsf{Dec}')$ of bounded width and constant depth *over the integers* from

[9] In this work, we will always assume that q is a prime. This means, we only study schemes over fields.

SKE. To do so, we need to delve a bit into polynomial interpolation by Vandermonde matrices. Formally, the Vandermonde matrix for univariate polynomials of degree d over the interpolation points $0, 1, \ldots, d$ is given by

$$V = \begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 1 & 1 & \cdots & 1 \\ 1 & 2 & 4 & 8 & \cdots & 2^d \\ 1 & 3 & 9 & 27 & \cdots & 3^d \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & d & d^2 & d^3 & \cdots & d^d \end{pmatrix} \in \mathbb{Z}^{(d+1)\times(d+1)}. \tag{71}$$

Now, if $c = (c_0, \ldots, c_d) \in \mathbb{R}^{d+1}$ is the coefficient vector of a polynomial $f(X) = \sum_{i=0}^{d} c_i X^i$, then c is mapped by the Vandermonde matrix to the vector of evaluations of f over the interpolation points, i.e.

$$V \cdot c = \begin{pmatrix} f(0) \\ \vdots \\ f(d) \end{pmatrix}. \tag{72}$$

This means, we can deduce the coefficients of f from its values by inverting V, i.e.,

$$\begin{pmatrix} c_0 \\ \vdots \\ c_d \end{pmatrix} = V^{-1} \cdot \begin{pmatrix} f(0) \\ \vdots \\ f(d) \end{pmatrix}. \tag{73}$$

But, there is a problem: the inverse V^{-1} of the Vandermonde matrix is not integer for $d > 1$. In particular, we cannot directly relate the inverse of V over $\mathbb{Z}_q$ with its real inverse. However, the polynomials we will study are all given over $\mathbb{Z}_q$. Fortunately, we can solve this problem by scaling V^{-1} by $d!$. As we will show in Section 2.2, the matrix $d! \cdot V^{-1}$ is always integer, even if V is the multivariate Vandermonde matrix. In particular, $d! \cdot V^{-1}$ is regular over $\mathbb{Z}_q$ and its entries are bounded by a constant.

Now, let $f(X) = \sum_{i=0}^{d} c_i X_i$ be a polynomial with coefficients $c_0, \ldots, c_d \in \mathbb{Z}_q$. Assume that $f(X)$ has small values at $0, \ldots, d$, i.e.

$$|f(x) \bmod q| \leq q/C \tag{74}$$

for $x = 0, \ldots, d$. By applying $d! \cdot V^{-1} \in \mathbb{Z}^{(d+1)\times(d+1)}$ on the vector $(f(0), \ldots, f(d))$, we see that the infinity norm of the scaled coefficient vector

$$d! \cdot \begin{pmatrix} c_0 \\ \vdots \\ c_d \end{pmatrix} = (d! \cdot V^{-1}) \cdot \begin{pmatrix} f(0) \\ \vdots \\ f(d) \end{pmatrix} \tag{75}$$

is bounded by $\left|\left|d! \cdot V^{-1}\right|\right|_\infty \cdot B$. The operator norm $\left|\left|d! \cdot V^{-1}\right|\right|_\infty$ of $d! \cdot V^{-1}$ is a constant that only depends on d. Since we have $B \leq q/C$, we can ensure that the scaled coefficients $d! \cdot c_0, \ldots, d! \cdot c_d$ are 'small enough' by setting C

appropriately. For each $i \in \{0, \ldots, d\}$, choose a small integer representation $a_i \in \{-(q-1)/2, \ldots, (q-1)/2\}$ s.t. $a_i \bmod q = d! \cdot c_i$. The polynomial

$$g(X) := \sum_{i=0}^{d} a_i \cdot X^i \in \mathbb{Z}[X] \tag{76}$$

can be seen as the integer interpretation of $d! \cdot f \in \mathbb{Z}_q[X]$. In particular, we have

$$g(X) \bmod q = d! \cdot f(X). \tag{77}$$

As we explained above, the coefficients of g are multiplicatively bounded by a constant from q (i.e., $a_i \leq q/C'$ for a large enough constant C'). Now, if we evaluate g at small inputs $x \in \{0, \ldots, 2d\}$, then the integer $g(x)$ is smaller than q. In particular, when evaluating $d! \cdot f(X)$ at x, no arithmetic reduction modulo q will happen. So, instead of evaluating $d! \cdot f(x)$ over $\mathbb{Z}_q$, we can evaluate $g(x)$ over $\mathbb{Z}$ and—up to a change of domains—obtain arithmetically the same result.

We will use this insight to construct an SKE scheme $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \mathsf{Dec}')$ over $\mathbb{Z}$ from the SKE scheme $\mathsf{SKE} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{Dec})$ over $\mathbb{Z}_q$. While ciphertexts of SKE are vectors in $\mathbb{Z}_q^m$, ciphertexts of SKE' will be vectors in $\mathbb{Z}^m$. If Enc is of depth d and width B over $\mathbb{Z}_q$, Enc' will be of depth d and width $d! \cdot B$ over $\mathbb{Z}$. From the construction, it will be clear that SKE' is correct and IND-CPA secure iff SKE is correct and IND-CPA secure, respectively.

SKE': On input 1^λ, SKE' samples $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$ and outputs $\mathsf{msk}' := \mathsf{msk}$ as master secret key.

Enc': On input msk' and a message $x \in \{0, \ldots, 2d\}$, Enc' samples $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)$ and computes

$$\mathsf{ct}' := d! \cdot \mathsf{ct}. \tag{78}$$

Enc' interprets ct' as integer vector (with small entries between $-(q-1)/2$ and $(q-1)/2$) and outputs it.

Dec': On input msk' and $\mathsf{ct}' \in \mathbb{Z}^m$, Dec' computes

$$\mathsf{ct} := \mathsf{ct}' \cdot (d!)^{-1} \bmod q. \tag{79}$$

Dec' outputs $\mathsf{Dec}(\mathsf{msk}', \mathsf{ct})$.

Since Enc' only applies an easily invertible operation on ciphertexts of Enc, we can directly see that SKE' is correct and secure if SKE is correct and secure. We argue that Enc' is of depth d if Enc is so. Let $\mathsf{Enc}_{\mathsf{off}}$ be the offline algorithm of Enc, and let us construct an offline algorithm $\mathsf{Enc}'_{\mathsf{off}}$ for Enc':

$\mathsf{Enc}'_{\mathsf{off}}$: On input msk', $\mathsf{Enc}'_{\mathsf{off}}$ samples $(r_1, \ldots, r_m) \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$. Note that each $r_j \in \mathbb{Z}_q[X]$ is a univariate polynomial of degree d over $\mathbb{Z}_q$.

$\mathsf{Enc}'_{\mathsf{off}}$ scales each r_j with $d!$ and interprets $d! \cdot r_j$ as an integer vector $r'_j \in \mathbb{Z}[X]$. It outputs $(r'_1, \ldots, r'_m)$.

As we explained above, for the small inputs $x \in \{0, \ldots, 2d\}$, the absolute values of $r'_1(x), \ldots, r'_m(x)$ will not become larger than $(q-1)/2$. In particular, the vector $(r'_1(x), \ldots, r'_m(x))$ will coincide with integer interpretation of $(d! \cdot r_1(x), \ldots, d! \cdot r_m(x))$. Hence, Enc' has the same depth as Enc.

We can now apply Informal Theorem 7: if we are given an SKE SKE of depth d and width $B \leq q/C$ over $\mathbb{Z}_q$, we can derive the SKE scheme SKE' over $\mathbb{Z}$ from it. SKE' is of depth d and width $d!\cdot B$ over $\mathbb{Z}$. Further, it has the same decryption probability as SKE. If $\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') = \mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec})$ is non-negligible, then the adversary $\mathcal{A}$ from Informal Theorem 7 has a non-negligible advantage against the IND-CPA security of SKE' while performing $\mathsf{poly}(\lambda + d!B) = \mathsf{poly}(\lambda + B)$ arithmetic operations. By the security reduction from SKE' to SKE, we get an indirect attack on SKE with the same advantage and same asymptotic time complexity class $\mathsf{poly}(\lambda + B)$. Hence, the claim of Informal Theorem 8 follows.

Lattice-Based FE

Let us finally turn to lattice-based functional encryption schemes. Remember that we required in Informal Definition 1 that a lattice-based FE scheme $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ for a functionality $\mathcal{F} : \mathbb{Z}_p^n \to \mathbb{Z}_p$ meets the following requirements:

1. Each ciphertext is a vector in $\mathbb{Z}_q^m$, and Enc is of constant depth d_1 over $\mathbb{Z}_q$.
2. Each secret key $\mathsf{sk} \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f)$ is a multivariate polynomial $\mathsf{sk} \in \mathbb{Z}_q[C_1, \ldots, C_m]$ of constant degree d_2.
3. Decryption works by polynomially evaluating the secret key at the ciphertext and rounding the result to the nearest integer in $\{0, \ldots, p-1\}$, i.e.

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = \left\lceil \frac{\mathsf{sk}(\mathsf{ct})}{\lceil q/p \rceil} \right\rfloor \in \{0, \ldots, p-1\}. \tag{80}$$

p is the message modulus of the scheme, and we can assume that it lies at least in $\omega(1)$, i.e., p is always larger than some constant.

Now, FE does almost fulfil the requirements of Informal Theorem 8: it is a (functional) encryption scheme of constant depth over $\mathbb{Z}_q$. However, its width is unbounded, i.e., the entries of ciphertexts of FE will usually spread over the whole field $\mathbb{Z}_q$. However, assuming that FE is perfectly correct, we can approach this problem as follows: let $x \in \mathbb{Z}_p^n$ be some message and let $f \in \mathcal{F}$ be a function that vanishes on x, i.e., $f(x) = 0$. Because of decryption correctness, we have for the ciphertext $\mathsf{ct}_x \leftarrow \mathsf{Enc}(\mathsf{msk}, x)$ and secret key $\mathsf{sk}_f \leftarrow \mathsf{Dec}(\mathsf{msk}, f)$

$$0 = \mathsf{Dec}(\mathsf{sk}_f, \mathsf{ct}_x) = \left\lceil \frac{\mathsf{sk}_f(\mathsf{ct}_x)}{\lceil q/p \rceil} \right\rfloor. \tag{81}$$

However, this can only be the case if $|\mathsf{sk}_f(\mathsf{ct}_x)|$ is smaller than $\frac{\lceil q/p \rceil}{2} \leq q/p \in o(q)$. In particular, the distribution of $\mathsf{sk}_f(\mathsf{ct}_x)$ is bounded by $B = q/C$ where C is the constant from Informal Theorem 8. Let us roll out this idea: fix some subspace $V \subset \mathbb{Z}_p^n$ and let $f_1, \ldots, f_Q \in \mathcal{F}$ be functions that vanish on V. Further, let $f_* \in \mathcal{F}$ be a meaningful function on V, i.e., f_* is not constant on V. Draw $\mathsf{sk}_i \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_i)$, for $i \in [Q]$, and consider the map

$$S : \mathbb{Z}_q^m \longrightarrow \mathbb{Z}_q^Q \tag{82}$$

$$c \longmapsto (\mathsf{sk}_1(c), \ldots, \mathsf{sk}_Q(c)). \tag{83}$$

S is of degree d_2. Denote by $S \circ \mathsf{Enc}$ the concatenation of S and Enc, i.e., we first generate a ciphertext $\mathsf{ct}_x \leftarrow \mathsf{Enc}(\mathsf{msk}, x)$ and, then, output $S(\mathsf{ct}_x)$. The composition $S \circ \mathsf{Enc}$ is of depth $d_1 \cdot d_2$, since the polynomials $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$ are fixed and of degree d_2. The map S resp. the secret keys $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$ will help us turn FE into an SKE scheme of constant depth and *bounded* width $o(q)$.

However, it remains to clarify how we can extract any relevant information of $x \in V$ from the small decryption noises $\mathsf{sk}_1(\mathsf{ct}_x), \ldots, \mathsf{sk}_Q(\mathsf{ct}_x)$. In general, this will be impossible: secure lattice-based FE schemes make sure that the noises $\mathsf{sk}_1(\mathsf{ct}_x), \ldots, \mathsf{sk}_Q(\mathsf{ct}_x)$ contain no information about x. However, in the complicated tasks of function-hiding and compact FE, this independence of noise can not be maintained. Indeed, we will in both cases show that the secret key $\mathsf{sk}_* \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_*)$ of a *meaningful* function $f_* \in \mathcal{F}$ will be (linearly and algebraically) related to the secret keys $\mathsf{sk}_1(\mathsf{ct}_x), \ldots, \mathsf{sk}_Q(\mathsf{ct}_x)$. This will allow us to decrypt a small portion of information, which will be large enough to invoke Informal Theorem 8.

Let us now sketch the SKE scheme $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \_)$ over $\mathbb{Z}_q$ that we derive from the lattice-based FE scheme FE. Our template strategy will have some gaps: we will not specify the decryption algorithm Dec', yet, and we will leave some open points in the setup algorithms. How exactly these gaps are filled will depend on the concrete functionality we will attack later.

Setup': On input 1^λ, Setup' samples $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$. Further, Setup' chooses a degree-1 map

$$\nu : \mathbb{Z}_p \longrightarrow \mathbb{Z}_p^n \tag{84}$$

and functions $f_1, \ldots, f_Q, f_* \in \mathcal{F}$ s.t. we have for all $x \in \mathbb{Z}_p$

$$\forall j \in [Q] : \; f_j(\nu(x)) = 0, \tag{85}$$

$$f_*(\nu(x)) = x. \tag{86}$$

For $j \in \{1, \ldots, Q, *\}$, Setup' samples

$$\mathsf{sk}_j \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_j). \tag{87}$$

Finally, Setup' outputs as new master secret key

$$\mathsf{msk}' := (\mathsf{msk}, \nu, \mathsf{sk}_1, \ldots, \mathsf{sk}_Q, \mathsf{sk}_*). \tag{88}$$

Enc': On input $\mathsf{msk}' = (\mathsf{msk}, \nu, \mathsf{sk}_1, \ldots, \mathsf{sk}_Q, \mathsf{sk}_*)$ and a message $x \in \{0, \ldots, 2d\}$, Enc' samples $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, \nu(x))$. It outputs the new ciphertext

$$\mathsf{ct}' := (\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_Q(\mathsf{ct})) \in \mathbb{Z}_q^Q. \tag{89}$$

As we explained, SKE' is of constant depth $d_1 \cdot d_2$ and bounded width q/p over $\mathbb{Z}_q$. We claim that SKE' is IND-CPA secure if FE is IND-CPA secure. Indeed, a reduction can simulate the IND-CPA security game for SKE' while playing the IND-CPA security game for FE. Since each function $f_1, \ldots, f_Q$ vanishes on each message $\nu(x)$, $x \in \{0, \ldots, 2d\}$, the reduction is allowed to query the secret keys $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$ from the challenger for the IND-CPA security of FE. Hence, the reduction can compute each output of Enc' by querying a ciphertext ct from its own challenger and applying $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$ on ct.

We can deduce from Informal Theorem 8 now the following theorem:

Informal Theorem 9 (Theorem 93)**.** *If there exists a decryption algorithm* Dec' *for* $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}, \_)$ *s.t. the decryption probability* $\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}')$ *is by a non-negligible amount larger than* $1/(2d+1)$*, then there exists an adversary that makes* $\mathsf{poly}(\lambda + q/p)$ *arithmetic operations and has a non-negligible advantage against the selective IND-CPA security of* FE*.*

Informal Theorem 9 simplifies our task a lot. In fact, in the case of function-hiding and compact FE schemes, we only need to specify the functions $f_1, \dots, f_Q$, f_*, ν chosen by Setup' and give a decryption algorithm Dec' that provably has non-negligible advantage at decryption. This is quite easy to accomplish in the context of function-hiding FE, as we will see in the following. In the context of compact FE, the problem will turn out to be a bit tricky. This is where algebraic relations will come into play.

Function-Hiding FE

A function-hiding FE scheme should ensure that an adversary, given a secret key for a function f and a ciphertext for a message x, learns nothing about f and x, except $f(x)$. Usually, this is captured by an extended IND-CPA security game where the adversary submits pairs of messages (x_0, x_1) and of functions (f_0, f_1). The challenger always answers with ciphertexts for x_b and secret keys for f_b, where b is the secret bit the adversary has to guess. The adversary then wins a run of this game when it guesses b correctly, and it is ensured that we have

$$f_0(x_0) = f_1(x_1) \tag{90}$$

for all key queries (f_0, f_1) and ciphertext queries (x_0, x_1).

In our case, it suffices to consider a weaker notion of function-hiding security where the adversary is not allowed to make any ciphertext queries. We will call $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ **function-hiding** secure if, with overwhelming probability over $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$, the distributions

$$(\mathsf{KeyGen}(\mathsf{msk}, f_i^{(0)}))_{i=1,\dots,Q} \qquad \text{and} \qquad (\mathsf{KeyGen}(\mathsf{msk}, f_i^{(1)}))_{i=1,\dots,Q} \tag{91}$$

are computationally indistinguishable for every pair of sequences $f_1^{(0)}, \dots, f_Q^{(0)}$, $f_1^{(1)}, \dots, f_Q^{(1)} \in \mathcal{F}$. Let us explain why this suffices: note that secret keys $\mathsf{sk} \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f)$ are polynomials of constant degree d_2 over m variables. Hence, all secret keys lie in the same vector space $\mathbb{Z}_q[C_1, \dots, C_m]^{\leq d_2}$ of polynomial dimension $\binom{m+d_2}{d_2}$. In particular, we can learn the space of secret keys for one specific function.

Fix a function $f \in \mathcal{F}$ and let $\mathsf{sk}_1, \dots, \mathsf{sk}_Q \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f)$ be Q identically distributed secret keys for f. By setting $Q > \binom{m+d_2}{d_2}$ large enough, we can ensure that we have

$$\Pr_{\mathsf{sk} \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f)}[\mathsf{sk} \in \operatorname{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1, \dots, \mathsf{sk}_Q\}] \geq 1 - o(1), \tag{92}$$

i.e., the next secret key we sample for f will lie with high probability in the space generated by the Q previously sampled secret keys. Indeed, this property must hold for each memoryless distribution of vectors over a space of polynomial dimension. In particular, this property is independent of the function f.

Now, set $f_1 = \ldots = f_Q = 0$ to be the zero function and let $f_*(X_1, \ldots, X_n) = X_1$ be the function that maps a message vector to its first coordinate. We claim that, if we draw $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q \leftarrow \mathsf{KeyGen}(\mathsf{msk}, 0)$ and $\mathsf{sk}_* \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_*)$, then we must have

$$\Pr[\mathsf{sk}_* \in \operatorname{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1, \ldots, \mathsf{sk}_Q\}] \geq 1 - o(1). \tag{93}$$

Indeed, if Eq. (93) would not hold, we would have found a simple test to distinguish between secret keys of 0 and f_*, which would break the function-hiding property of FE. Let $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \_)$ be the secret-key encryption scheme for FE where Setup' chooses

$$\nu : \mathbb{Z}_p \longrightarrow \mathbb{Z}_p^n \tag{94}$$

$$x \longmapsto (x, 0, \ldots, 0) \tag{95}$$

and $f_1 = \ldots = f_Q = 0$ and f_* as above. Remember that the new master secret key output by Setup' consists of

$$\mathsf{msk}' := (\mathsf{msk}, \nu, \mathsf{sk}_1, \ldots, \mathsf{sk}_Q, \mathsf{sk}_*) \tag{96}$$

where $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$ are secret keys for $f_1, \ldots, f_Q$, i.e., for the zero function, and sk_* is a secret key for f_*. Note that, since f_* maps each vector to its first coordinate, we have

$$f_*(\nu(x)) = x. \tag{97}$$

Now, according to Informal Theorem 9, there can be no decryption algorithm for SKE' with non-negligible decryption advantage, as long as FE is IND-CPA secure. However, consider the following decryption algorithm:

Dec': Dec' receives $\mathsf{msk}' = (\mathsf{msk}, \nu, \mathsf{sk}_1, \ldots, \mathsf{sk}_Q, \mathsf{sk}_*)$ and ct' as input. Remember that ct' is of the shape

$$\mathsf{ct}' = (\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_Q(\mathsf{ct})) \in \mathbb{Z}_q^Q. \tag{98}$$

Now, Dec' checks if we have

$$\mathsf{sk}_* \in \operatorname{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1, \ldots, \mathsf{sk}_Q\}. \tag{99}$$

If that is not the case, then Dec' outputs a random element of $\{0, \ldots, 2d\}$ and terminates.

However, if FE is function-hiding, then Eq. (99) will hold with high probability. In this case, Dec' computes scalars $\alpha_1, \ldots, \alpha_Q \in \mathbb{Z}_q$ s.t.

$$\mathsf{sk}_* = \alpha_1 \cdot \mathsf{sk}_1 + \ldots + \alpha_Q \cdot \mathsf{sk}_Q. \tag{100}$$

Denote the entries of ct' by $(c_1, \ldots, c_Q)$. Dec' computes

$$c := \alpha_1 \cdot c_1 + \ldots + \alpha_Q \cdot c_Q \tag{101}$$

and outputs

$$\left\lceil \frac{c}{\lceil q/p \rceil} \right\rfloor. \tag{102}$$

We claim that Dec' has a high probability to decrypt a ciphertext correctly. Draw $\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda)$ and $\mathsf{ct}' = (\mathsf{sk}_1(\mathsf{ct}), \dots, \mathsf{sk}_Q(\mathsf{ct})) \leftarrow \mathsf{Enc}'(\mathsf{msk}', \nu(x))$ for some arbitrary $x \in \{0, \dots, 2d\}$. Since FE is function-hiding and $Q > \binom{m+d}{d}$ a large enough polynomial, sk_* will lie in the span generated by the polynomials $\mathsf{sk}_1, \dots, \mathsf{sk}_Q$ with high probability. In this case, Dec' will find scalars $\alpha_1, \dots, \alpha_Q$ s.t.

$$\mathsf{sk}_* := \alpha_1 \cdot \mathsf{sk}_1 + \ldots + \alpha_Q \cdot \mathsf{sk}_Q. \tag{103}$$

Hence, we have for the value c computed by Dec'

$$\begin{aligned} c =& \alpha_1 \cdot c_1 + \ldots + \alpha_Q \cdot c_Q && (104)\\ =& \alpha_1 \cdot \mathsf{sk}_1(\mathsf{ct}) + \ldots + \alpha_Q \cdot \mathsf{sk}_Q(\mathsf{ct}) && (105)\\ =& (\alpha_1 \cdot \mathsf{sk}_1 + \ldots + \alpha_Q \cdot \mathsf{sk}_Q)(\mathsf{ct}) = \mathsf{sk}_*(\mathsf{ct}). && (106) \end{aligned}$$

It follows that Dec' outputs with high probability the value

$$\left\lceil \frac{c}{\lceil q/p \rceil} \right\rfloor = \left\lceil \frac{\mathsf{sk}(\mathsf{ct})}{\lceil q/p \rceil} \right\rfloor = \mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) \tag{107}$$

which is $f_*(\nu(x)) = x$, since FE is perfectly correct. Hence, SKE' has a decryption probability of $1 - o(1)$, which is by a non-negligible amount larger than $1/(2d+1)$. Informal Theorem 9 postulates now that FE cannot be IND-CPA secure if it is correct, function-hiding secure and lattice-based in our sense.

Compact FE

Finally, let us turn to compact functional encryption schemes. Concretely, we consider a lattice-based FE scheme $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ for the functionality

$$\mathcal{F} = \{f \in \mathbb{Z}_p[X_1, \dots, X_n] \mid \deg f \le 2\} \tag{108}$$

of polynomials of degree 2. This functionality can be trivially achieved by taking a lattice-based FE scheme for linear functions and relinearizing each degree-2 polynomial to a linear function over $\binom{n+2}{2}$ variables. However, the resulting FE scheme will have ciphertexts of length $\Omega(n^2)$. Hence, we need to restrict the size of ciphertexts.

Let each ciphertext of FE be a vector in $\mathbb{Z}_q^m$. We will require that ciphertexts of FE are **(relaxed) compact**, i.e., there is a constant $e > 0$ such that we have for the length m of ciphertexts

$$m \in O(n^{2-e}). \tag{109}$$

In other words, we require that the length of ciphertexts is by a small polynomial factor smaller than the trivially achievable ciphertext length of $\Omega(n^2)$.

We claim that compactness gives us an attack surface on which we can apply algebraic relations. For $i, j \in [n]$, $i < j$, consider the monomial functions

$$f_{i,j}(X) := X_i \cdot X_j \in \mathcal{F} \tag{110}$$

and their secret keys

$$\mathsf{sk}_{i,j} \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_{i,j}). \tag{111}$$

The secret keys $(\mathsf{sk}_{i,j})_{i,j}$ are all polynomials in $\mathbb{Z}_q[C_1, \ldots, C_m]$ of degree $\leq d_2$. The number of these secret keys is $\binom{n}{2}$, making it quadratic in n, while the number m of their variables is subquadratic due to the compactness of ciphertexts. This implies again the existence of algebraic relations among the secret keys $(\mathsf{sk}_{i,j})_{i,j}$. Set

$$Q := \binom{n}{2} \in \Theta(n^2) = \Theta(m^{1+e/(2-e)}) \tag{112}$$

and let $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$ be an enumeration of the secret keys $\mathsf{sk}_{i,j}$, $1 \leq i < j \leq n$. Concretely, Informal Theorem 1 implies the existence of a polynomial $h \in \mathbb{Z}_q[S_1, \ldots, S_Q]$ with the following properties:

$$h \neq 0, \tag{113}$$

$$h(\mathsf{sk}_1, \ldots, \mathsf{sk}_Q) = 0, \tag{114}$$

$$\deg h \leq D \in O(m^{1-\frac{e}{(2-e)(d_2-1)}}) \in o(m). \tag{115}$$

The fact that the degree of h is sublinear in m is not of great importance here. It will suffice that $\deg h$ is smaller than the message modulus p. Now, an important property of the function collection $f_1, \ldots, f_Q$ is that we can turn one of them *on* while turning the others *off*. Let us explain this: pick any function $f_k = X_i \cdot X_j$ from the collection, and choose a value $z \in \mathbb{Z}_p$. We want to find a point x s.t. each f_ℓ vanishes on x, except f_k, which is supposed to evaluate to z on x. Given the simple nature of our functions, we can set x to be the point that is z at position i, 1 at position j and 0 at all remaining positions. Now, we must have for $\ell \in [Q]$

$$f_\ell(x) = \begin{cases} 0, & \text{if } \ell \neq k, \\ z, & \text{if } \ell = k. \end{cases} \tag{116}$$

Indeed, if $\ell \neq k$, then f_ℓ must contain one variable, which is neither X_i nor X_j and, therefore, evaluate to zero. On the other hand, $f_k = X_i \cdot X_j$ evaluates to $z \cdot 1$ at x. Now, h relates the secret keys $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$ of the functions $f_1, \ldots, f_Q$. Let $x \in \mathbb{Z}_p^n$ be the point from above and let $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)$ be its corresponding ciphertext. For $\ell \neq k$, each evaluation $\mathsf{sk}_\ell(\mathsf{ct})$ must be bounded, since $f_\ell(x) = 0$. However, $\mathsf{sk}_k(\mathsf{ct})$ must contain non-trivial information about x, since $f_k(x) = z$. This allows us to decrypt z when given $(\mathsf{sk}_\ell(\mathsf{ct}))_{\ell \neq k}$ as follows: if we plug in each $\mathsf{sk}_\ell(\mathsf{ct})$, $\ell \neq k$, into $h(S_1, \ldots, S_m)$, we get a univariate polynomial

$$g(S_k) := h(\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_{k-1}(\mathsf{ct}), S_k, \mathsf{sk}_{k+1}(\mathsf{ct}), \ldots, \mathsf{sk}_m(\mathsf{ct})) \in \mathbb{Z}_q[S_k] \tag{117}$$

with sublinear degree $\deg g \leq \deg h \leq D \in o(m)$. Let us assume[10] that g is not the zero polynomial. Note that the value of interest $\mathsf{sk}_k(\mathsf{ct})$ must be a root of g, since we have

$$g(\mathsf{sk}_k(\mathsf{ct})) = h(\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_m(\mathsf{ct})) = (h(\mathsf{sk}_1, \ldots, \mathsf{sk}_m))(\mathsf{ct}) = 0(\mathsf{ct}) = 0. \tag{118}$$

However, as a non-zero polynomial of degree D, g has at most D different roots in $\mathbb{Z}_q$. Hence, we can restrict $\mathsf{sk}_k(\mathsf{ct})$ (and the value $z \in \mathbb{Z}_p$ to which it decrypts)

[10] This is a non-trivial assumption as we will see in this work.

to a set of at most D candidates. If D is significantly smaller than p, then this gives us a non-negligible advantage at guessing z. Indeed, if z is any value in $\mathbb{Z}_p$, then the trivial probability to guess z correctly (without any auxiliary information) is $1/p$. However, with the auxiliary information $(\mathsf{sk}_\ell(\mathsf{ct}))_{\ell \neq k}$, we can reduce the choices for z to the set

$$\left\{ \left\lceil \frac{w}{\lceil q/p \rceil} \right\rfloor \;\middle|\; w \in \mathbb{Z}_q, g(w) = 0 \right\}, \tag{119}$$

whose cardinality is bounded by D. Hence, with the auxiliary information $(\mathsf{sk}_\ell(\mathsf{ct}))_{\ell \neq k}$, we get the guessing probability $1/D$, which is by a non-negligible amount better than $1/p$ if $D \in \mathsf{poly}(\lambda)$ and $D \leq p/2$.

In the following, we will use the above observation to construct an SKE scheme $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \mathsf{Dec}')$ that is derived by the compact lattice-based scheme FE. We will use the polynomial h (and hence the compactness of FE) to persuade us of a non-negligible decryption advantage of Dec'. Unfortunately, our arguments will have flaws (g may be zero), which we can only fix in the case where h is of constant degree. Let us first sketch the scheme $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \mathsf{Dec}')$:

Setup' : On input 1^λ, Setup' samples $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$. It computes an enumeration

$$f_1, \ldots, f_Q \in \mathcal{F} \tag{120}$$

of all monomials $X_i X_j$, $1 \leq i < j \leq n$, where $Q = \binom{n}{2}$. Without loss of generality, we assume that the last function f_Q is the product of the first two variables, i.e., $f_Q = X_1 \cdot X_2$. Setup' computes a linear map

$$\nu : \mathbb{Z}_p \longrightarrow \mathbb{Z}_p^n \tag{121}$$

$$z \longmapsto (z, 1, 0, \ldots, 0) \tag{122}$$

that maps a number $z \in \mathbb{Z}_p$ to a vector whose first coordinate is z and whose second coordinate is 1 (the remaining coordinates are set to 0). Setup' computes for each $k \in [Q]$ a secret key

$$\mathsf{sk}_k \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_k) \tag{123}$$

and outputs the new master secret key

$$\mathsf{msk}' := (\mathsf{msk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_Q, \nu). \tag{124}$$

Enc': On input the master secret key

$$\mathsf{msk}' = (\mathsf{msk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_Q, \nu) \tag{125}$$

and a message $z \in \mathbb{Z}_p$, Enc' samples $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, \nu(z))$ and outputs the new ciphertext

$$\mathsf{ct}' := (\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_{Q-1}(\mathsf{ct})) \in \mathbb{Z}_q^{Q-1}. \tag{126}$$

Dec': On input $\mathsf{msk}' = (\mathsf{msk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_Q, \nu)$ and $\mathsf{ct}' = (c_1, \ldots, c_{Q-1})$, Dec' computes a non-zero algebraic relationship $h \in \mathbb{Z}_q[S_1, \ldots, S_Q]$ of $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$ of minimal degree. It sets

$$g(S_Q) := h(c_1, \ldots, c_{Q-1}, S_Q) \in \mathbb{Z}_q[S_Q] \tag{127}$$

and computes the set

$$W := g^{-1}(0) = \{s \in \mathbb{Z}_q \mid g(s) = 0\}. \tag{128}$$

Finally, it draws $s \leftarrow W$ uniformly at random and outputs

$$\left\lceil \frac{s}{\lceil q/p \rceil} \right\rfloor. \tag{129}$$

Now, Informal Theorem 9 states either FE is not IND-CPA secure or Dec' does not have non-negligible advantage at decryption. We want to argue that the probability $\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}')$ that Dec' decrypts a ciphertext correctly is by a non-negligible amount larger than $\frac{1}{p}$.

If $\mathsf{ct}' = (\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_{Q-1}(\mathsf{ct})) \leftarrow \mathsf{Enc}'(\mathsf{msk}', \nu(z))$ is a ciphertext of a message z and if

$$g(S_Q) = h(\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_{Q-1}(\mathsf{ct}), S_Q) \tag{130}$$

is not the zero polynomial, then $W = g^{-1}(0)$ contains at most $\deg g \leq D$ many elements. One of those elements must be $\mathsf{sk}_Q(\mathsf{ct})$. With probability $\geq 1/D$, Dec' will choose $\mathsf{sk}_Q(\mathsf{ct})$ from W and output the correct message

$$\left\lceil \frac{\mathsf{sk}_Q(\mathsf{ct})}{\lceil q/p \rceil} \right\rfloor = f_Q(\nu(z)) = z. \tag{131}$$

D is sublinear in m, hence, if p is at least as large as m it follows

$$\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') - \frac{1}{p} \geq \frac{1}{D} - \frac{1}{p} = \frac{p - D}{pD} \in \omega\left(\frac{1}{m}\right). \tag{132}$$

However, there is a critical gap in our reasoning. What happens if the polynomial g of Eq. (130) is zero at decryption? In particular, what can we do if g is with overwhelming probability zero at decryption? This problem will turn out to be quite resistant. In fact, we can only solve it in the cases where the degree bound D of h is constant, for example, where $m \in O(n)$ is linear and all secret keys $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$ are quadratic over $\mathbb{Z}_q$. In our solution, we will have to compromise: instead of an algebraic relation h, we will search for a polynomial $\widetilde{h} \in \mathbb{Z}_q[S_1, \ldots, S_Q]$, which will in a non-negligible number of cases fulfil

$$\widetilde{h}(\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_{Q-1}(\mathsf{ct}), \mathsf{sk}_Q(\mathsf{ct})) = 0, \tag{133}$$

$$\widetilde{h}(\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_{Q-1}(\mathsf{ct}), S_Q) \neq 0. \tag{134}$$

This will allow us to correctly decrypt the message z with non-negligible probability. Unfortunately, we can prove the existence of such a polynomial $\widetilde{h} \in \mathbb{Z}_q[S_1, \ldots, S_Q]$ only in the cases where D is bounded by a constant. Hence, our lower bound for compact FE holds currently only for cases where $m \in O(n^\delta)$ and $\deg \mathsf{sk} = d_2 > 1$ with $\delta \cdot d_2$ being exactly the degree of the supported functionality.

Limits on Our Lower Bounds

Let us finish by discussing some restrictions on our lower bounds. In particular, we will discuss techniques that are not captured by our framework for lattice-based functional encryption schemes, given in Informal Definition 1. These techniques may have the potential to circumvent the results of this work and achieve compact or function-hiding FE schemes.

On Parameters. First note that we always assume that the outer modulus q of the lattice-based schemes we study is prime. Usually, lattice-based schemes are not bound to any specific modulus, and the LWE problem is assumed to be hard even if the modulus q factorizes into very small prime factors. Extending our results to non-prime moduli q can turn out to be tricky, since we heavily borrow techniques from the realm of linear algebra. On the other hand, any lattice-based FE scheme that circumvents our results here by making use of a non-prime modulus q would need to exploit the prime factorization of q. However, for a lattice-based scheme this seems to be very untypical.

The second parameter restriction we have is given by the bound

$$B \in O(q/p) \tag{135}$$

where p is the modulus of the message space of the FE scheme. The SKE schemes we construct are of width B, hence, the resulting IND-CPA adversary uses a mean square distinguisher that needs to query $\mathsf{poly}(\lambda + B)$ ciphertexts and perform $\mathsf{poly}(\lambda + B)$ arithmetic operations over $\mathbb{Z}$. Now, one can attempt to circumvent our lower bounds here by choosing q/p exponentially large, for example. However, even in that case, our adversary would outperform direct attacks on LWE of comparable parameters (for example, the attack of Arora and Ge [AG11] would have a time complexity of $2^{\Theta\left((q/p)^2\right)}$ if we assume that the noise size lies in $\Theta(q/p)$). Hence, we think that—even in the case that q/p were to be exponential—our attack would give a clear indication that the resulting FE scheme cannot be proven secure under learning with errors.

On Ring-LWE and Module-LWE. Ring- and Module-LWE are popular variants of LWE, in which one allows for the extra structure of the ring extension $\mathbb{Z}_q \subset R_q := \mathbb{Z}_q[X]/(X^n + 1)$. Both variants will not help at circumventing our lower bounds here. The reason is that the operations over R_q can be simulated by polynomials over $\mathbb{Z}_q$, since the extension $\mathbb{Z}_q \subset R_q$ is finite.

On Bit Decomposition. Bit decomposition or, more generally, inverse Gadget sampling [MP12] are popular techniques in lattice-based cryptosystems, that are necessary for lattice-based fully homomorphic encryption schemes. Decomposing an element of $\mathbb{Z}_q$ into its bits is an operation of very high algebraic degree. In particular, our lattice-based FE model does not cover the case that the encryption or decryption algorithms decompose messages or ciphertexts into their bits.

As we pointed out in the introduction, there are attribute-based and predicate encryption schemes [Bon+14; GVW15] that rely on fully homomorphic encryption and are proven secure under LWE. Hence, we think that fully homomorphic encryption resp. inverse gadget sampling might propose a technique to circumvent our lower bounds completely. However, up to now, it is not clear how this technique might be used in the context of functional encryption.

On Binary Messages. In all our lower bounds on lattice-based FE schemes, we require that the message modulus p is larger than some constant that depends on the scheme. In particular, if p equals 2, then all messages of an FE scheme $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ would be binary, and the requirement for Enc to be of constant depth would be trivially satisfied. However, the proofs for our lower bounds would not hold in that case. Hence, one may ask if we can achieve a function-hiding or compact FE scheme with binary messages. In the context of function-hiding FE, we can simplify the question even more and ask for the existence of a lattice-based *one-bit multiplication scheme*, i.e., a scheme where one can encrypt single bits in left and right ciphertexts and evaluate a multiplication once when given a left and a right ciphertext. Concretely, we repeat here the following question from [TÜ23]:

Question 2. *Let $\mathcal{X} = \{0,1\}$ and $\mathcal{F} = \{f_0, f_1\}$ where the functions $f_0, f_1 : \mathcal{X} \to \mathcal{X}$ are given by*

$$f_0(x) := 0 \qquad \text{and} \qquad f_1(x) = x. \tag{136}$$

Note that the functionality $\mathcal{F} : \mathcal{X} \to \mathcal{X}$ essentially computes a logical AND.

We ask if there is a symmetric function-hiding IND-CPA secure correct FE scheme $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ *for $\mathcal{F} : \mathcal{X} \to \mathcal{X}$ s.t.* KeyGen *and* Enc *output vectors in $\mathbb{Z}_q^m$ and decryption works by*

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = \begin{cases} 0, & \text{if } \left|\mathsf{sk}^T \cdot \mathsf{ct}\right| \leq B, \\ 1, & \text{if } \left|\mathsf{sk}^T \cdot \mathsf{ct}\right| > B, \end{cases}$$

for some threshold $B < q/2$?

We note that, since $\mathcal{X}$ and $\mathcal{F}$ only contain two elements, we ask here if there are keyed distributions $\mathcal{E}_{\mathsf{msk},0}$, $\mathcal{E}_{\mathsf{msk},1}$, $\mathcal{S}_{\mathsf{msk},0}$, $\mathcal{S}_{\mathsf{msk},1}$ over $\mathbb{Z}_q^m$ s.t. we have for $a, b \in \{0,1\}$ and $\mathsf{ct} \leftarrow \mathcal{E}_{\mathsf{msk},a}$, $\mathsf{sk} \leftarrow \mathcal{S}_{\mathsf{msk},b}$

$$\left|\mathsf{sk}^T \cdot \mathsf{ct}\right| \text{ is large iff } a \cdot b = 1,$$

and s.t. additionally a poly-time adversary cannot distinguish between $\mathcal{E}_{\mathsf{msk},0}$ and $\mathcal{E}_{\mathsf{msk},1}$, when given access to $\mathcal{S}_{\mathsf{msk},0}$, and between $\mathcal{S}_{\mathsf{msk},0}$ and $\mathcal{S}_{\mathsf{msk},1}$, when given access to $\mathcal{E}_{\mathsf{msk},0}$.

On Double Arithmetic Reduction. As we explained before, our Informal Definition 1 covers FE schemes that perform an arithmetic reduction at decryption instead of a rounding operation. I.e., our results also apply to FE schemes where decryption works by

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = (\mathsf{sk}(\mathsf{ct}) \bmod q) \bmod p \in \{0, \ldots, p-1\}. \tag{137}$$

In fact, we will show in Lemma 53

$$(\mathsf{sk}(\mathsf{ct}) \bmod q) \bmod p = 0 \implies (p^{-1} \cdot \mathsf{sk}(\mathsf{ct}) \bmod q) < \frac{q}{2p}. \tag{138}$$

However, as far as we know, we cannot show a similar statement when two or more arithmetic reductions are applied at decryption.

The quadratic FE scheme of Agrawal and Rosen [AR17], for example, computes at decryption

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = ((\mathsf{sk}(\mathsf{ct}) \bmod q) \bmod p_1) \bmod p_2 \in \{0, \ldots, p_2 - 1\} \tag{139}$$

for prime moduli $q > p_1 > p_2$. Another example is given by the noisy FE scheme of Agrawal and Pellet-Mary [AP20]. Hence, applying arithmetic reductions multiple times at decryption might help at achieving compact or function-hiding lattice-based FE.

On Algebraically Weaker Notions of Security. Finally, when studying our approach for showing lower bounds for compact lattice-based FE, one notices that we heavily rely on asking secret keys for a lot of algebraically dependent functions. This is fine, since our adversary adheres to the rule of the corresponding IND-CPA game of the FE scheme (confer Game 3 for a formal definition). However, it might be that the usual IND-CPA security for FE is too revealing in the lattice-based setting. Let us motivate this: again, assume that $\mathcal{F}$ is the functionality of quadratic polynomials. Let us play the IND-CPA game of an FE scheme for $\mathcal{F}$ with a corresponding challenger, and assume that we ask the challenger for secret keys sk_1 for the functions $f_1 := X_1$ and sk_2 for $f_2 := X_2X_1$. Now, the function $f_3 := X_2$ is algebraically dependent from f_1, f_2, since we have $f_3 = f_2/f_1$. However, according to the rules of the normal IND-CPA game, the secret keys $\mathsf{sk}_1, \mathsf{sk}_2$ may not help us at distinguishing ciphertexts ct_0 and ct_1 of $x^{(0)} = (0, \ldots, 0)$ and $x^{(1)} = (0, 1, \ldots, 0)$, respectively. Now, assume that the secret keys $\mathsf{sk}_1, \mathsf{sk}_2$ behave somewhat similar to the functions f_1, f_2 s.t. $\mathsf{sk}_2/\mathsf{sk}_1$ would give a somewhat correct secret key sk_3' for the function f_3. In the case of noise-free decryption, this would yield no problem, since $\mathsf{sk}_1(\mathsf{ct}_0) = 0 = \mathsf{sk}_1(\mathsf{ct}_1)$ would always be zero, which would make the formula $\mathsf{sk}_3'(\mathsf{ct}_b) = \mathsf{sk}_2(\mathsf{ct}_b)/\mathsf{sk}_1(\mathsf{ct}_b)$ useless. However, in the noisy decryption setting, it is very unlikely that $\mathsf{sk}_1(\mathsf{ct}_b)$ is exactly zero. This may make the formula $\mathsf{sk}_3'(\mathsf{ct}_b) = \mathsf{sk}_2(\mathsf{ct}_b)/\mathsf{sk}_1(\mathsf{ct}_b)$ useful and, hence, open an inevitable attack vector for an adversary.

A potential countermeasure against this type of attacks would be to weaken the notion of IND-CPA security and to make it more considerate with regard to algebraic dependencies. For this end, let the message modulus p be a prime and let us introduce some technicalities: for a set of polynomials $f_1, \ldots, f_Q \in \mathbb{Z}_p[X_1, \ldots, X_n]$, denote by $\mathbb{Z}_p[f_1, \ldots, f_Q]$ the $\mathbb{Z}_p$-algebra generated by $f_1, \ldots, f_Q$. I.e., $\mathbb{Z}_p[f_1, \ldots, f_Q]$ is the smallest subring of $\mathbb{Z}_p[X]$ that contains $\mathbb{Z}_p$ and each f_j. For short, we write $\mathbb{Z}_p[F] := \mathbb{Z}_p[f_1, \ldots, f_Q]$. We define the *algebraic closure* of the ring $\mathbb{Z}_p[F]$ by

$$\overline{\mathbb{Z}_p[F]} := \{g \in \mathbb{Z}_p[X] \mid \exists h \in \mathbb{Z}_p[F][T] : h \neq 0 \land h(g) = 0\}. \tag{140}$$

I.e., $\overline{\mathbb{Z}_p[F]}$ contains each $g \in \mathbb{Z}_p[X]$ that is algebraically dependent of the polynomials $f_1, \ldots, f_Q$, in the sense that there is a non-zero univariate polynomial $h \in \mathbb{Z}_p[F][T]$ with coefficients in $\mathbb{Z}_p[F]$ that vanishes at g. Note that T is a fresh new variable.

We can now formally state a security game for FE schemes that is conceptually weaker than the usual IND-CPA Game 3 for FE schemes. This security game has the same interactions and phases than the normal IND-CPA game, however, it requires the adversary to be more careful at function and encryption queries.

Game 1 (Algebraically Restricted Selective IND-CPA Security Game)**.** Let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be an FE scheme for a function space $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$.

We define the **algebraically restricted selective IND-CPA security game** of FE as the following game between a stateful challenger $\mathcal{C}$ and a stateful adversary $\mathcal{A}$:

Phase 1: On input 1^λ, the adversary $\mathcal{A}$ sends two lists $(x_i^{(0)})_{i=1}^N, (x_i^{(1)})_{i=1}^N \in \mathcal{X}_\lambda^N$ to the challenger $\mathcal{C}$. Additionally, the adversary sends a list of functions $(f_i)_{i=1}^Q \in \mathcal{F}_\lambda^Q$ to the challenger.

Phase 2: The challenger $\mathcal{C}$ receives as input the unary encoded security parameter 1^λ and collects the lists $(x_i^{(0)})_{i=1}^N, (x_i^{(1)})_{i=1}^N \in \mathcal{X}_\lambda^N$ and $(f_j)_{j=1}^Q \in \mathcal{F}_\lambda^Q$ from the adversary. The challenger draws a random bit $b \leftarrow \{0,1\}$, and samples a fresh master secret key $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$. It encrypts all messages of $(x_i^{(b)})_{i=1}^N$, i.e., it computes for $i = 1, \ldots, N$

$$\mathsf{ct}_i := \mathsf{Enc}(\mathsf{msk}, x_i^{(b)}). \tag{141}$$

Further, it generates secret keys for all functions submitted by the adversary, i.e., it computes for $j = 1, \ldots, Q$

$$\mathsf{sk}_j := \mathsf{KeyGen}(\mathsf{msk}, f_j). \tag{142}$$

Finally, the challenger sends the list of ciphertexts $(\mathsf{ct}_i)_{i=1}^N$ and the list of secret keys $(\mathsf{sk}_j)_{j=1}^Q$ to the adversary.

Phase 3: Upon receiving $(\mathsf{ct}_i)_{i=1}^N$ and $(\mathsf{sk}_j)_{j=1}^Q$, the adversary $\mathcal{A}$ does some computations on its own and finally responds with a guess $b' \in \{0,1\}$.

The adversary $\mathcal{A}$ wins a run of the above game if it guesses the bit b of the challenger correctly, i.e., $b = b'$. *Additionally, we require that each function g that would help at distinguishing a ciphertext pair $(x_i^{(0)}, x_i^{(1)})$ is algebraically independent of all secret key queries $f_1, \ldots, f_Q$. In other words, we require that we have for each $g \in \overline{\mathbb{Z}_p[f_1, \ldots, f_Q]}$ and $i \in [N]$*

$$g(x_i^{(0)}) = g(x_i^{(1)}). \tag{143}$$

We think it makes sense to first study the feasibility of compact lattice-based FE schemes that are secure in the sense of Game 1, before turning to the question of compact lattice-based FE schemes that are secure in the normal IND-CPA security game.

Related Work

Before we begin with the technical parts of this work, let us view some related literature to understand the novelty and relevancy of the results of this work:

On Algebraic Independency in the Context of Cryptography

Algebraic (in)dependency has already been studied several times in the context of algebraic and arithmetic computability. For example, Dvir, Gabizon, and Wigderson [DGW07] used a result of Wooley [Woo96] to show that—over a large enough field[11]—the output distribution of algebraically independent polynomials $f_1, \ldots, f_m$ (when evaluated at uniformly random seeds) is close to m independent random variables.

Applebaum, Avron, and Brzuska [AAB15] studied the predictability problem of arithmetic circuits. Oversimplifying, they showed that the output of a polynomial $f_0 \in k[X]$ at a point $x \in k^n$ is predictable (in an information-theoretical sense) given the evaluations of $f_1, \ldots, f_m$ at $x \in k^n$ iff the derivative $\nabla f_0(x)$ evaluated at a random point $x \leftarrow k^n$ lies with high probability in the vector space spanned by $\nabla f_1(x), \ldots, \nabla f_m(x)$ (again, we assume that k is very large). There is a beautiful connection between this observation and restriction given by algebraic relations, which is rooted in a canonical interplay between relations and differentials. Let us sketch, how this connection works: first, note that we have the following equivalency

$$\Pr_{x \leftarrow k^n} [\nabla f_0(x) \in \operatorname{span}_k\{\nabla f_1(x), \ldots, \nabla f_m(x)\} \subseteq k^n] \geq 1 - o(1) \tag{144}$$

$$\iff \nabla f_0 \in \operatorname{span}_{k(X)}\{\nabla f_1, \ldots, \nabla f_m\} \subseteq k(X)^n. \tag{145}$$

Now, let us assume that f_0 is algebraic over $k(f_1, \ldots, f_m)$. Then, there is an algebraic relation $h \in k[Y_0, \ldots, Y_m]$ of minimal degree[12] with the following properties:

$$h(f_0, \ldots, f_m) = 0, \tag{146}$$

$$\frac{\partial h}{\partial Y_0} \neq 0. \tag{147}$$

The last inequality ensures that Y_0 appears non-trivially in h. Since $h(f_0, \ldots, f_m)$ is constantly zero, its derivative must vanish. Hence, we have

$$0 = \nabla(h(f_0, \ldots, f_m)) = \sum_{i=0}^{m} \frac{\partial h}{\partial Y_i}(f_0, \ldots, f_m) \cdot \nabla f_i. \tag{148}$$

Since $\frac{\partial h}{\partial Y_0} \neq 0$, ∇f_0 lies in the $k(f_0, \ldots, f_m)$-vector space spanned by $\nabla f_1, \ldots, \nabla f_m$. Hence, if f_0 is algebraically dependent of $f_1, \ldots, f_m$ over k then ∇f_0 is linearly dependent of $\nabla f_1, \ldots, \nabla f_m$ over $k(X)$. The converse direction is true, too, but a bit tricky to prove: assume that

$$\nabla f_0 \in \operatorname{span}_{k(X)}\{\nabla f_1, \ldots, \nabla f_m\} =: V. \tag{149}$$

Without loss of generality, we assume that $m = n - 1$, and that the functions $f_1, \ldots, f_{n-1}, X_n$ give us a transcendency basis for the field extension $k \subset k(X)$.

[11] The requirement on the field being large is crucial. For example, the polynomials $f_1 = X$ and $f_2 = XY$ are algebraically independent. However, over $\mathbb{Z}_2$, the values (x, xy) (for $x, y \leftarrow \{0, 1\}$) are clearly distinguishable from two independently and uniformly random bits.

[12] We assume here again that the characteristic of k is large enough and that the extension $k(f_1, \ldots, f_m) \subseteq k(f_0, f_1, \ldots, f_m)$ is separable.

Then, it is clear that f_0 is algebraic over $f_1, \ldots, f_{n-1}, X_n$. In particular, there is an algebraic relationship $h \in k[Y_0, \ldots, Y_n]$ of minimal degree with the following properties:

$$h(f_0, f_1, \ldots, f_{n-1}, X_n) = 0, \tag{150}$$

$$\frac{\partial h}{\partial Y_0} \neq 0. \tag{151}$$

We claim that Y_n does not appear in h. More formally, we want to show $\frac{\partial h}{\partial Y_n} = 0$. Again, we have

$$0 = \nabla(h(f_0, \ldots, f_{n-1}, X_n)) \tag{152}$$

$$= \sum_{i=0}^{n-1} \frac{\partial h}{\partial Y_i}(f_0, \ldots, X_n) \cdot \nabla f_i + \frac{\partial h}{\partial Y_n}(f_0, \ldots, X_n) \cdot \nabla X_n. \tag{153}$$

Because of Eq. (149), the first n summands of the sum on the right-hand side lie in the vector space V. ∇X_n cannot lie in V, since $f_1, \ldots, f_{n-1}, X_n$ form a transcendency basis. Hence, the sum can only be zero if the last summand $\frac{\partial h}{\partial Y_n}(f_0, \ldots, X_n) \cdot \nabla X_n$ vanishes. Ergo, $\frac{\partial h}{\partial Y_n}(f_0, \ldots, X_n)$ must be zero. We assumed that under all polynomials fulfilling Eqs. (150) and (151), h is chosen of minimal degree. Hence, $\frac{\partial h}{\partial Y_n}$ cannot fulfil Eq. (151). It follows that $\frac{\partial h}{\partial Y_n}$ only depends on $Y_1, \ldots, Y_n$. However, the polynomials $f_1, \ldots, f_{n-1}, X_n$ are transcendent, ergo $\frac{\partial h}{\partial Y_n}(f_0, \ldots, X_n)$ can only be zero if $\frac{\partial h}{\partial Y_n}(Y_1, \ldots, Y_n)$ is zero. Hence, Y_n does not appear non-trivially in h. Therefore, h is actually an algebraic relation of the polynomials $f_0, \ldots, f_{n-1}$.

In total, the following statements are equivalent (for a large enough field):

$$\Pr_{x \leftarrow k^n}\left[\nabla f_0(x) \in \mathrm{span}_k\{\nabla f_1(x), \ldots, \nabla f_m(x)\} \subseteq k^n\right] \geq 1 - o(1) \tag{154}$$

$$\iff \nabla f_0 \in \mathrm{span}_{k(X)}\{\nabla f_1, \ldots, \nabla f_m\} \subseteq k(X)^n \tag{155}$$

$$\iff \exists h \in k[Y_0, \ldots, Y_m] : \; h \neq 0, \; h(f_0, \ldots, f_n) = 0, \; \frac{\partial}{\partial Y_0} h \neq 0. \tag{156}$$

On Attacks On Algebraic PRGs

We will discuss here other attacks on algebraic PRGs of constant degree. Important will be the class of Macaulay matrix-based attacks whose time complexity we will compare with the time complexity of attacks of this work.

Parts of the following have been taken mostly verbatim from my previous work [Üna23a], where I already compared relation-based PRG attacks with other algebraic attacks, and underwent minor modifications. Since local PRGs are not a focus of this work, we will not discuss attacks that are specially tailored for them. For comparison with special attacks for PRGs of low locality, I refer the reader to my previous works [Üna23c; Üna23a].

Relinearization Attacks. Each known attack on PRGs of constant degree over arbitrarily large fields is of algebraic nature. A first approach is to understand the equation $F(X) = y$ as a polynomial equation system with n variables $X_1, \ldots, X_n$ and m polynomial equations $f_1(X) = y_1, \ldots, f_m(X) = y_m$. Relinearizing this equation system yields a linear equation system, on which one can apply Gaussian elimination. If we have enough equations, i.e. $m \geq \binom{n+\deg F}{\deg F}$, then with high probability [AG11] this linear equation system can be solved for a possible seed x, or at least the satisfiability of the linear equation system can be checked. This leads to a basic attack on algebraic PRGs that is efficient and very reliable (its advantage is provably noticeable). This attack can already be improved: we don't need that m is greater than $\binom{n+\deg F}{\deg F}$, in fact, it suffices that $m \in \Omega(n^{\deg F})$. If m is smaller than $\binom{n+\deg F}{\deg F}$, but has the same asymptotic complexity then it suffices to populate the linear equation system with more polynomial equations that can be generated from $F(X) = y$ up to some constant degree.

Macaulay Matrices and Groebner Bases. Extending the idea of the relinearization-and-elimination algorithm above leads to Groebner basis-based, or rather Macaulay matrix-based, attacks. Groebner bases together with a first algorithm for computing them have been introduced by Buchberger [Buc76]. Faster algorithms have been given by Faugère [Fau99; Fau02]. Additionally, the XL-algorithm with a lot of variations [CKPS00; CCNY12; DBMMW08; MMDB08; YC05] has been introduced. These algorithms are based on Macaulay matrices [Mac16; Laz83]. Their core idea is to solve the polynomial equation system $F(X) = y$ by computing a Groebner basis for the ideal $(f_1(X) - y_1, \dots, f_m(X) - y_m) \subset k[X]$ for some monomial ordering. Most algorithms do this by computing a Macaulay matrix up to an increasing degree and applying Gaussian elimination on it: the Macaulay matrix for degree D is the matrix where each row represents a polynomial $X^\alpha \cdot (f_i(X) - y_i)$, for a multi-index α with $||\alpha||_1 \leq D - \deg f_i$, and where each column represents a monomial of $k[X]$ up to degree D. I.e., the rows of the Macaulay matrix are the coefficient vectors of polynomials $X^\alpha \cdot (f_i(X) - y_i)$. The columns are ordered according to the monomial ordering. By applying Gaussian elimination to the Macaulay matrix of degree D one can extract a Groebner basis from it, if D is large enough. In most cases, the Groebner basis will be of the shape $\{X_1 - x_1, \dots, X_n - x_n\}$, which allows to directly read off the solution $X = x \in k^n$ of the polynomial equation system $F(X) = y$. Hence, Macaulay matrix-based attacks are usually inversion attacks that try to extract the seed $X = x$ from the PRG problem $F(X) = y$.

While Macaulay matrix-based algorithms perform well in practice, it is hard to give formal guarantees for them. Dubé [Dub90] showed that, in the worst case, the highest degree of polynomials of a reduced Groebner basis for an equation system $F(X) = y$ is doubly exponential. However, the doubly exponential degree only occurs in extreme cases. On average, the maximum degree for which a Macaulay matrix must be computed is suspected to be upper-bounded by the degree of regularity (in the case of graded anti-lexicographic monomial orders [CG21; CG23]). The degree of regularity is a popular heuristic for Groebner basis-based algorithms, it has been shown to lie in $O(n^{1-e/(d-1)})$ for a system of $m \geq n^{1+e}$ equations of degree d [Üna23c]. This would yield an inversion attack of suspected time complexity $n^{O(n^{1-e/(d-1)})}$.

In the case of refutation, better bounds can be given: a Macaulay matrix-based algorithm for refutation problems only checks if the equation $1 = 0$ can be deduced at sufficiently high degree. If up to some degree the span of the rows of the Macaulay matrix does not contain a vector that corresponds to a constant non-zero polynomial, then the algorithm assumes that the system $F(X) = y$ is solvable and decides that y lies in the image of the PRG F. Otherwise, the algorithm could prove that $F(X) = y$ is unsatisfiable and refutes y. Our results here will show that it suffices to compute the Macaulay matrix up to some degree in $O(n^{1-e/(d-1)})$ if the base field is large enough. For small fields, it suffices to compute the Macaulay matrix up to some degree in $O(\log(n)^{1/(d-1)} \cdot n^{1-e/(d-1)})$ to refute an equation system $F(X) = y$ of degree d with high probability over the randomness of $y \leftarrow k^m$.

Distinguishing Based on Algebraic Relations. Using algebraic relations (sometimes also incorrectly called *annihilating polynomials*) instead of Macaulay matrices to distinguish true randomness from polynomial pseudorandomness is somewhat recent. For example, Miles, Sahai, and Zhandry [MSZ16] used algebraic relations to analyse the multilinear maps of Garg, Gentry, and Halevi [GGH13], and, in his master thesis, Zichron [Zic17] studied algebraic relations on polynomial PRGs of constant degree over large fields and gave lower bounds for their degree.

The Barrier of Applebaum and Lovett. Unfortunately, the time complexity of algebraic algorithms must be subexponential, in general. In fact, Applebaum and Lovett [AL16] proved that—even in the context of local PRGs—the time complexity of an algebraic algorithm deciding if $y \in \{0,1\}^{n^{1+e}}$ lies in the image of a random local function $F : \{0,1\}^n \to \{0,1\}^{n^{1+e}}$ is lower-bounded by $n^{O(n^{1-16 \cdot e/(d-1)})}$ where d is the *rational degree*[13] of the predicate of F.

It is an interesting open problem to construct new algebraic algorithms that perform provably faster than the barrier of Applebaum and Lovett and have non-negligible advantage. Note that such new algorithms must avoid computing Macaulay matrices.

The Polynomial Method. In a recent line of work [LPTWY17; BKW19; Din21b; Din21a], worst-case algorithms for finding solutions of a polynomial equation system $F(X) = y$ have been given. While these algorithms all have an exponential runtime of $2^{\Theta(n)}$, they significantly beat the typical brute-force search for a solution x by an exponential speed-up. For solving quadratic systems over $\mathbb{Z}_2$ with n variables, the first algorithm has been given by Lokshtanov, Paturi, Tamaki, Williams, and Yu [LPTWY17] with a worst-case runtime of $O(2^{0.8765n})$. Their algorithm has been improved to have an asymptotic runtime of $\mathsf{poly}(n) \cdot 2^{0.804n}$ by Björklund, Kaski, and Williams [BKW19]. Dinur [Din21b] further improved this asymptotically to $O(2^{0.6943n})$ and gave a better algorithm for concrete parameters [Din21a].

At the heart of all of those algorithms is the *polynomial method*, which is usually used to prove lower bounds in circuit complexity theory [Wil14]. The idea is that—instead of checking if $f_i(x) = y_i$ for each $i \in [m]$ separately on an

[13] The *rational degree* of a predicate $P : \{0,1\}^d \to \{0,1\}$ is defined as the smallest number e s.t. there exist polynomials $Q, R \in \mathbb{Z}_2[X_1, \ldots, X_d]$ of degree e that fulfil $P(X) \cdot Q(X) = R(X) \bmod (X_1^2 - X_1, \ldots, X_d^2 - X_d)$ and $Q \neq 0$. In other words, P can be written as the rational function $P(X) = \frac{R(X)}{Q(X)}$ of degree e whenever Q does not evaluate to zero.

input $x \in \{0,1\}^n$—we consider the polynomial

$$p(X) := \prod_{i=1}^{m} (1 + f_i(X) - y_i) - 1 \in \mathbb{Z}_2[X]. \tag{157}$$

$p(X)$ does vanish on input $x \in \{0,1\}^n$ iff each equation $f_i(x) = y_i$ is fulfilled. However, the degree of p is very large, so instead of looking for roots of p, one uses a randomized simplification $\widetilde{p}$ of p with significantly lower degree. By using various algebraic tricks, it is possible to iterate fast over $x \in \{0,1\}^n$ and check if $\widetilde{p}$ vanishes on x.

Comparing Runtimes. As we explained, Macaulay matrix- and relation-based algebraic attacks on PRGs are connected and our upper bounds for relation-based attacks can be carried over to Macaulay matrix-based attacks. This raises the question which class of attacks is faster. As we will see in Section 1.5, this depends on the attack environment.

Let $F : k^n \to k^m$ be a PRG of degree d and stretch $m \geq n^{1+e}$ over a field k. Macaulay matrix-based distinguishing attacks on the equation system $F(X) = y$ have a time complexity of $O\Big(\binom{n+d}{d} \cdot \binom{m+D}{D}^2\Big)$ for $D \in O(n^{1-e/(d-1)})$. Computing an algebraic relation h for F has a time complexity of $O\Big(\binom{m+D}{D}^3\Big)$, however, this step can be preprocessed. When we know F ahead of time (for example, when F is some fixed public PRG), then we can compute h in an offline phase without knowing y. The time complexity of the online phase, where we have to apply h on y, has a time complexity of $O\Big(D \cdot \binom{m+D}{D}\Big)$. As we can see in the overview of Table 2, Macaulay matrix-based attacks are faster when F is not known ahead of time. However, in the preprocessing setting, attacks based on algebraic relations outperform Macaulay matrix-based attacks.

Runtime	Algebraic Relations	Macaulay Matrices
Offline	$O\Big(\binom{m+D}{D}^3\Big)$	0
Online	$O\Big(D \cdot \binom{m+D}{D}\Big)$	$O\Big(\binom{n+d}{d} \cdot \binom{m+D}{D}^2\Big)$

Table 2: An overview of estimated upper bounds on the time complexity of algebraic attacks on a PRG $F : k^n \to k^m$ of degree d. $D \in \mathbb{N}$ is chosen minimal with $\binom{m+D}{D} > \binom{n+dD}{dD}$.

On Lower Bounds for Lattice-Based Functional Encryption

Beside the lower bounds given in this work, there are in fact no other lower bound results for the special case of lattice-based functional encryption. However, there are some generic results on the security of IBE, ABE and FE. Further, since we tried to describe here a framework in which we proved our lower bounds, we will revisit some cryptographic models and frameworks in which lower bounds have been proven.

Lower Bounds for Functional Encryption and Relatives

De Caro, Iovino, Jain, O'Neill, Paneth, and Persiano [De +13] and Agrawal, Gorbunov, Vaikuntanathan, and Wee [AGVW13] showed that any FE scheme that supports the functionality of some weak pseudorandom function family cannot be simulation secure (in the sense that there is a simulator that can simulate ciphertexts and secret keys by only knowing the expected evaluation values of messages and functions). Their core observation is that a successful simulator would need to break pseudorandomness of the weak pseudorandom function.

Lewko and Waters [LW14] showed that each *straight-line* black-box reduction of a hardness assumption to the adaptive security of *checkable* hierarchical IBE schemes must have an exponential security loss. They proved this by the use of *meta-reductions*, which were introduced by Coron [Cor02]. Recently, Brakerski and Medina [BM23] extended this result to rewinding reduction proofs for the adaptive security of ABE schemes.

Cryptographic Models

The Random Oracle Model. The random oracle model (ROM), introduced by Bellare and Rogaway [BR93], is one of the first idealized models in the cryptographic setting. In the ROM, all parties have black-box access to the same truly random, but deterministic functionality $H : \{0,1\}^* \to \{0,1\}^n$. The function H idealizes hash functions and other similar primitives and can be thought as a substitute for secure hash algorithms in the real world.

The ROM can be very useful for proving the security of various cryptographic primitives. For example, the security of signature schemes is usually proven in the ROM with varying degrees of how much the reduction may influence the functionality H.

The Generic Group Model. The generic group model (GGM) is a popular model, in which lower and upper bounds for group-based assumptions and primitives can be shown. In the GGM, all parties have access to oracles that perform group operations of $\mathbb{Z}_p$. However, instead of receiving the elements of $\mathbb{Z}_p$ directly, actors in the GGM are given encodings of elements of $\mathbb{Z}_p$. In the GGM of Shoup [Sho97], those encodings are random bit strings, while, in the GGM of Maurer [Mau05], those encodings are just an increasing enumeration of register addresses. It is important to note that an actor in the GGM can always ask a corresponding oracle if two encodings point to the same element of $\mathbb{Z}_p$.

While the GGM has initially been deployed to test the plausibility of group-based hardness assumptions and give lower bounds for the time complexity of generic attacks against them, it is nowadays feasible to prove the security of whole cryptosystems in the GGM, if no reduction to group-based assumptions of polynomial size is apparent.

The Algebraic Group Model. The algebraic group model (AGM), introduced by Fuchsbauer, Kiltz, and Loss [FKL18], is supposed to be an intermediate model between the GGM and the standard model[14]. Note that in the GGM

[14] The *standard model* is the normal world of Turing machines without access to any oracles.

the capabilities of adversaries and reductions are limited to purely *generic* operations. I.e., in the GGM an adversary or reduction is not permitted to perform any operation that is specially tailored to the representation of group elements.

The AGM model tries to relax this condition by restricting an adversary resp. a reduction to be *algebraic*. I.e., whenever it outputs a group element, it must additionally output an explanation, i.e., a degree-1 or degree-2 polynomial that relates the exponent of the outputted group element to the exponents of all group elements received by the adversary resp. reduction.

Since the AGM is not fully well-defined, there is a lot of confusion about its feasibility and usefulness [ZZK22; Zha22].

On Lattice-Based NIKE. Guo, Kamath, Rosen, and Sotiraki [GKRS20] studied the feasibility of lattice-based non-interactive key exchange (NIKE). They invoked a framework, which is more rigid than our lattice-based FE framework, where two actors send LWE samples $A \cdot x_1 + e_1$ and $A^T \cdot x_2 + e_2$ as their key parts, respectively. After exchanging key parts, both parties may apply a *reconciliation* function to derive a common secret key. The authors could show lower bounds for the complexity for the used reconciliation function and the correctness of the lattice-based NIKE schemes. Recently, Langrehr [Lan23] extended those results to the multi-user setting, and showed, in a malicious model, infeasibility and feasibility for LWE with polynomial and superpolynomial modulus-to-noise rates, respectively.

Arithmetic Cryptography. Applebaum, Avron, and Brzuska [AAB15] introduced the model of arithmetic cryptography and showed several upper and lower bounds in this framework. Intuitively, an arithmetic algorithm computes an algebraic functionality, which is given in a form that is independent of the field over which the algorithm is cast. More precisely, an *arithmetic* circuit has generic field elements as input, output and intermediate values. It may apply any of the four basic arithmetic field operations, call the constants 0 and 1, perform zero-checks and sample random bits and field elements. It is crucial, that the description of the arithmetic circuit is independent of any field. In fact, an arithmetic circuit can be evaluated over any field of finite size, and it is expected to stay secure and correct independently of the concrete field.

Note that, in our framework, the online part of encryption algorithms and the first part of decryption algorithms are computed by polynomials of constant degree. Hence, these aspects fall into the class of arithmteic circuits. In fact, Applebaum, Avron, and Brzuska [AAB15] illustrate at the start of their text a lower bound for the communication complexity of a simple protocol for three (semi-arithmetic) parties, which has parallels to our treatment of lattice-based function-hiding FE. Let us first sketch a simplified version of their lower bound: let k be a (large enough) field. We consider two parties, Alice and Bob, who have secret data $x \in k$ and $y, z \in k^n$, respectively. Additionally, in a preprocessing phase, both parties were allowed to share common randomness and do some non-arithmetic computations on their own. Now, both parties want to send messages to a third party, Carol, such that Carol can evaluate the function

$$f(x, y, z) := y + x \cdot z \tag{158}$$

without learning anything else about x, y and z. The important requirement is that Carol is fully arithmetic, while Alice is arithmetic in the online phase of the protocol. I.e., in the preprocessing phase, Alice computed a polynomial map $\phi_A : k \to k^m$ and has to send $\phi_A(x)$ as message to Carol. Bob, on the other hand, does not need to be arithmetic and simply sends a polynomial map $\phi_B : k^m \to k^n$ as message to Carol. Carol is fully arithmetic and can only apply ϕ_B at $\phi_A(x)$. Correctness implies that we must have the following equality of polynomials[15]

$$\phi_B(\phi_A(X)) = f(X, y, z) = y + X \cdot z \tag{159}$$

for all possible inputs y, z of Bob. By deriving the above equality, we get

$$z = \frac{\nabla}{\nabla X} f(X, y, z) = \frac{\nabla}{\nabla X}(\phi_B(\phi_A(X))) = \nabla \phi_B(\phi_A(X)) \cdot \frac{\nabla}{\nabla X} \phi_A(X). \tag{160}$$

Carol is not aware of ϕ_A or $\frac{\nabla}{\nabla X}\phi_A$. However, she knows ϕ_B and can, in particular, compute its derivation $\nabla \phi_B(\phi_A(x))$ at Alice's message $\phi_A(x)$. Can this leak non-trivial information about the data z of Bob? Note that $\nabla \phi_B(\phi_A(x))$ is a matrix of shape $n \times m$. Carol knows that z must lie in the image of $\nabla \phi_B(\phi_A(x))$. If $m \geq n$ and $\nabla \phi_B(\phi_A(x))$ has full rank, then this does not leak any information about z. However, if $m < n$, then Carol can restrict z to a proper subspace of k^n. Hence, it follows that, in this semi-arithmetic setting, Alice must send at least $m \geq n$ field elements to Carol to compute the functionality $f(X, y, z) = y + X \cdot z$ without Carol learning anything non-trivial about the data of Bob.

We want to point out the relationship of this example to our lower bound for lattice-based function-hiding FE. Let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be an FE scheme for affine linear functions. For simplicity, we assume that all messages are of dimension 1. We require that FE is lattice-based, i.e., there is an offline algorithm $\mathsf{Enc}_{\mathsf{off}}$ that receives as input the master secret key msk and outputs m polynomials $r_1, \ldots, r_m \in \mathbb{Z}_q[X]$ s.t. ciphertexts of a message $x \in \mathbb{Z}_p$ are computed by

$$\mathsf{ct} := (r_1(x), \ldots, r_m(x)). \tag{161}$$

Further, secret keys $\mathsf{sk} \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f)$ are required to be polynomials $\mathsf{sk} \in \mathbb{Z}_q[C_1, \ldots, C_m]$ over m variables, and decryption works by computing

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = \left\lceil \frac{\mathsf{sk}(\mathsf{ct})}{\lceil q/p \rceil} \right\rfloor. \tag{162}$$

Now, we could use FE to build a three-party protocol as above and then could try to apply the lower bound of Applebaum, Avron, and Brzuska [AAB15] to show the insecurity of FE. Let us explain this: in a preprocessing phase, Alice and Bob sample a master secret key $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$. Additionally, Alice samples encryption randomness $(r_1, \ldots, r_m) \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$. In the online phase of the protocol, Alice computes $\mathsf{ct} := (r_1(x), \ldots, r_m(x))$ and sends ct to Carol. Let $n > m$. Bob, who does not need to be arithmetic, samples for each $i \in [n]$ a secret key $\mathsf{sk}_i \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_i)$ for the affine linear function

$$f_i(X) := y_i + X \cdot z_i \tag{163}$$

[15] At this point, we need that the size of k is larger than $\deg \phi_B \cdot \deg \phi_A$.

where $y = (y_1, \ldots, y_n)$ and $z = (z_1, \ldots, z_n)$. Finally, Bob sends the keys $\mathsf{sk}_1, \ldots, \mathsf{sk}_n$ to Carol. Let us denote the function, sent by Bob, by $\phi_B := (\mathsf{sk}_1, \ldots, \mathsf{sk}_n)$. Carol receives ct and ϕ_B and computes

$$\phi_B(\mathsf{ct}) = (\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_n(\mathsf{ct})). \tag{164}$$

Now, if $\mathsf{sk}_i(\mathsf{ct})$ would equal $f_i(x)$, then we could simply invoke the lower bound of Applebaum, Avron, and Brzuska [AAB15] and be done. This is, because the function-hiding IND-CPA security of FE implies that Carol may not learn anything about x, y and z except $xy + z$.

However, here is the crux: the decryption algorithm Dec is only semi-arithmetic. After computing $\phi_B(\mathsf{ct}) \in \mathbb{Z}_q$, Carol needs to round each coordinate from $\mathbb{Z}_q$ down to $\mathbb{Z}_p$ to obtain the final result. The problem[16] is that the rounding function $\lceil \cdot \rfloor$ is highly non-arithmetic, since its degree grows with the size of the field $\mathbb{Z}_q$. Hence, the arithmetic model alone can not arrive at the lower bound for function-hiding FE scheme that we give here. The difference between the arithmetic model and our approach is that we allow one operation of geometric nature at the end of decryption. By doing so, we can transfer a bit of algebraic cryptanalysis to the geometrical world of lattices. However, this comes at the cost of very tedious attacks and analyses.

[16] The problem could be solved if one could show that $\mathsf{sk}_i(r_1(X), \ldots, r_m(X))$ must be equal to $e_i + \lceil q/p \rceil \cdot (y_i + X \cdot z_i)$ for some fixed noise value e_i. However, one can not prevent that higher powers of X appear in a negligible way in $\mathsf{sk}_i(r_1(X), \ldots, r_m(X))$ such that these higher powers get rounded away at decryption.

On Notations and Algorithms

Let us start the technical part of this work by making some common ground and introducing definitions and conventions that are in effect over the whole text.

Notations

Sets. We denote by $\cup, \cap$ and $\setminus$ the usual set-theoretic operations. Note that we will denote the cardinality of a set in this text by $\#$. I.e., $\#S$ denotes the number of elements of the set S. We denote by $\emptyset = \{\}$ the empty set.

Logic. We denote by $\vee, \wedge$ and $\neg$ the usual operations from propositional logic. By $\forall$ and $\exists$, we denote the usual "For all"- and "For one"-quantifiers from first-order logic. Additionally, we will use the "For almost all"-quantifier $\forall_\infty$ and the "For infinitely many"-quantifier $\exists_\infty$. When given a set S and a predicate $\phi : S \to \{\mathsf{FALSE}, \mathsf{TRUE}\}$, both quantifiers are specified by

$$\forall_\infty x \in S : \ \phi(x) \iff \#\{x \in S \mid \neg\phi(x)\} < \infty, \tag{165}$$
$$\exists_\infty x \in S : \ \phi(x) \iff \#\{x \in S \mid \phi(x)\} = \infty. \tag{166}$$

Numbers. We denote by $\mathbb{N} = \{1, 2, 3, \ldots\}$ the natural numbers and by $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$ the natural numbers with zero. $\mathbb{Z}$ shall denote the integers, $\mathbb{Q}$ the rational and $\mathbb{R}$ the real numbers. Additionally, $\mathbb{R}_{>0} := \{r \in \mathbb{R} \mid r > 0\}$ and $R_{\geq 0} = \{r \in \mathbb{R} \mid r \geq 0\}$ shall denote the set of positive and non-negative reals.

For $n \in \mathbb{N}$, we set $[n] := \{1, \ldots, n\}$. For $a, b \in \mathbb{R}$ with $a \leq b$, we denote the closed, open and the two half-open intervals by

$$[a, b] = \{x \in \mathbb{R} \mid a \leq x \leq b\}, \tag{167}$$
$$(a, b) = \{x \in \mathbb{R} \mid a < x < b\}, \tag{168}$$
$$[a, b) = \{x \in \mathbb{R} \mid a \leq x < b\}, \tag{169}$$
$$(a, b] = \{x \in \mathbb{R} \mid a < x \leq b\}. \tag{170}$$

For $x \in \mathbb{R}_{>0}$, denote by $\log x$ the logarithm to base 2, i.e., we have

$$2^{\log x} = x. \tag{171}$$

Binomial Coefficients. For $n, k \in \mathbb{N}_0$, we denote the **binomial coefficient** *n choose k* by

$$\binom{n}{k} := \begin{cases} 0, & \text{if } k > n, \\ \prod_{i=0}^{k-1} \frac{n-i}{k-i}, & \text{if } 0 < k \leq n, \\ 1, & \text{if } k = 0. \end{cases} \tag{172}$$

Note that $\binom{n}{k}$ is always a non-negative integer. Further, we have the following inequalities for $n \in \mathbb{N}, k \in \{1, \ldots, n\}$

$$\left(\frac{n}{k}\right)^k \leq \binom{n}{k} \leq \left(e \cdot \frac{n}{k}\right)^k, \tag{173}$$

where e denotes Euler's number.

Rounding. Given $r \in \mathbb{R}$, we define the following rounding functions

$$\lceil r \rceil := \min \{z \in \mathbb{Z} \mid r \leq z\}, \tag{174}$$

$$\lfloor r \rfloor := \max \{z \in \mathbb{Z} \mid r \geq z\}, \tag{175}$$

$$\lceil r \rfloor := \max \left\{ z \in \mathbb{Z} \;\middle|\; |z - r| \leq \frac{1}{2} \right\}. \tag{176}$$

Finite Rings and Fields. For $q \in \mathbb{N}$, we denote by $\mathbb{Z}_q := \mathbb{Z}/q\mathbb{Z}$ the residue classes of integers modulo q. It is known that $\mathbb{Z}_q$ is a field iff q is prime. Note that finite fields are—up to ring isomorphisms—uniquely determined by their cardinality.

Vector Norms. For real vectors, we will denote by $||\cdot||_2$, $||\cdot||_1$ and $||\cdot||_\infty$ the euclidean, one- and infinity **norm**. I.e., for $x \in \mathbb{R}^n$, we set

$$||x||_2 := \sum_{i=1}^{n} x_i^2, \tag{177}$$

$$||x||_1 := \sum_{i=1}^{n} |x_i|, \tag{178}$$

$$||x||_\infty := \max_{i=1,\ldots,n} |x_i|. \tag{179}$$

For $M \in \mathbb{R}^{m \times n}$ and $p \in \{1, 2, \infty\}$, we define the corresponding **matrix norm** by

$$||M||_p := \max_{x \in \mathbb{R}^n \setminus \{0\}} \frac{||Mx||_p}{||x||_p}. \tag{180}$$

Polynomial Rings and Ideals. Given $n \in \mathbb{N}$ indeterminates $X_1, \ldots, X_n$ and a **multi-index** $\alpha \in \mathbb{N}_0^n$, we will set $X^\alpha := X_1^{\alpha_1} \cdots X_n^{\alpha^n}$ to be the monomial in the variables $X_1, \ldots, X_n$ with powers indexed by α. Given a ring R, we define the polynomial ring over R in the variables $X_1, \ldots, X_n$ by

$$R[X_1, \ldots, X_n] := \left\{ \sum_{\alpha \in \mathbb{N}_0^n} c_\alpha \cdot X^\alpha \;\middle|\; \forall_\infty \alpha \in \mathbb{N}_0^n : c_\alpha = 0 \right\}. \tag{181}$$

Usually, we will write $R[X]$ instead of $R[X_1, \ldots, X_n]$ when it is clear that we abbreviate the list of variables $X_1, \ldots, X_n$ by the symbol X. Given polynomials $f_1, \ldots, f_m \in R[X], m \in \mathbb{N}$, we denote by $R[f_1, \ldots, f_m]$ the smallest subring of $R[X]$ that contains R and the elements $f_1, \ldots, f_m$ and by $(f_1, \ldots, f_m) \subset R[X]$ the smallest ideal of $R[X]$ that contains $f_1, \ldots, f_m$.

If k is a field, then $k[X_1, \ldots, X_n]$ is a domain. In this case, we will denote by

$$k(X) := k(X_1, \ldots, X_n) := \left\{ \frac{f}{g} \;\middle|\; f, g \in k[X], g \neq 0 \right\} \tag{182}$$

the **function field** of n variables over k. For $f_1, \ldots, f_m \in k(X), m \in \mathbb{N}$, we denote by $K(f_1, \ldots, f_m)$ the smallest subfield of $k(X)$ that contains k and $f_1, \ldots, f_m$.

Degree and Graduations. In a polynomial ring $R[X] = R[X_1, \ldots, X_n]$, we will assign degrees to variables, monomials and polynomials. If not stated otherwise, the variable X_i has degree 1 i.e. $\deg X_i = 1$ for all $i \in [n]$. The **total degree** of a monomial $X^\alpha, \alpha \in \mathbb{N}_0^n$, is given by $\deg X^\alpha := \alpha_1 \cdot \deg X_1 + \ldots + \alpha_n \cdot \deg X_n$. Usually, we will just say **degree** when we speak of the total degree. The total degree of a polynomial $f(X) = \sum_{\alpha \in \mathbb{N}_0^n} c_\alpha \cdot X^\alpha \in R[X]$ is the maximum of the total degrees of all monomials that appear with a non-zero coefficient in f. Formally, for $f \neq 0$, we set

$$\deg \left(\sum_{\alpha \in \mathbb{N}_0^n} c_\alpha \cdot X^\alpha \right) := \max \{ \deg(X^\alpha) \mid c_\alpha \neq 0 \} \tag{183}$$

and for the zero polynomial we agree on $\deg(0) := 0$.

The polynomial ring $R[X]$ admits a **graduation** according to the total degree. Given $d \in \mathbb{N}_0$, we denote by $R[X]^d := \operatorname{span}_R \{ X^\alpha \mid d = ||\alpha||_1 \}$ the R-module that is generated by all monomials of degree exactly d. We then have the following isomorphism of R-modules

$$R[X] \cong \bigoplus_{d=0}^{\infty} R[X]^d. \tag{184}$$

We call a polynomial $f \in R[X]$ **homogenous** iff $f \in R[X]^{\deg f}$. Finally, to keep notation simple, we denote by $R[X]^{\leq d} := \operatorname{span}_R \{ X^\alpha \mid d \geq ||\alpha||_1 \}$ the R-module of all polynomials of $R[X]$ whose total degree is at most d.

If k is a field and each variable $X_1, \ldots, X_n$ has degree 1, then the vector space dimensions of $k[X]^d$ and $k[X]^{\leq d}$ are well known. In fact, they are given by

$$\dim_k k[X]^d = \binom{n+d-1}{d} \qquad \text{and} \qquad \dim_k k[X]^{\leq d} = \binom{n+d}{d}. \tag{185}$$

Asymptotic Behaviour. In this work, we will use the Bachmann-Landau notation to describe the asymptotic behaviour of functions from $\mathbb{N}$ to $\mathbb{R}$. Concretely, given $f : \mathbb{N} \to \mathbb{R}$, we set

$$\begin{aligned} O(f) :=& \{g : \mathbb{N} \to \mathbb{R} \mid \exists a > 0\ \forall_\infty n \in \mathbb{N} :\ g(n) \leq a \cdot f(n)\}, && (186)\\ o(f) :=& \{g : \mathbb{N} \to \mathbb{R} \mid \forall \varepsilon > 0\ \forall_\infty n \in \mathbb{N} :\ g(n) \leq \varepsilon \cdot f(n)\}, && (187)\\ \Omega(f) :=& \{g : \mathbb{N} \to \mathbb{R} \mid f \in O(g)\}, && (188)\\ \omega(f) :=& \{g : \mathbb{N} \to \mathbb{R} \mid f \in o(g)\}, && (189)\\ \Theta(f) :=& O(f) \cap \Omega(f). && (190) \end{aligned}$$

By abuse of notation, we will sometimes use the above classes in formulas, to imply bounds on the growth of involved terms. For example, for $f, g, h : \mathbb{N} \to \mathbb{R}$ and $c \neq 0$ the statement

$$f \in g + c \cdot O(h) \tag{191}$$

is equivalent to $\frac{f-g}{c} \in O(h)$. Similarly,

$$f \in 2^{\omega(g)} \tag{192}$$

means $\log(f) \in \omega(g)$. For $R \in \{\mathbb{N}, \mathbb{N}_0, \mathbb{Z}, \mathbb{Q}, \mathbb{R}, \mathbb{R}_{>0}\}$ and $f : \mathbb{N} \to \mathbb{R}$, we set

$$\mathsf{poly}_R(f) := \big\{g : \mathbb{N} \to R \bigm| \exists d \in \mathbb{N} :\ g \in O(f^d)\big\}. \tag{193}$$

For convenience, we will omit the subscript if R equals $\mathbb{N}$, i.e. $\mathsf{poly}(f) := \mathsf{poly}_{\mathbb{N}}(f)$. Further, we define the set $\mathsf{negl}(f)$ by

$$\mathsf{negl}(f) := \bigcap_{d \in \mathbb{N}} o(f^{-d}). \tag{194}$$

Usually, in this work, we will describe a function implicitly by the term it evaluates to. For example, by λ we will usually denote the identity $\mathbb{N} \to \mathbb{N}, \lambda \mapsto \lambda$ and by n^2, for example, we denote the quadratic function $\mathbb{N} \to \mathbb{N}, n \mapsto n^2$.

Now, let $p : \mathbb{N} \to \mathbb{R}$ be a function that describes the probability of some event. We call p **negligible** if $p \in \mathsf{negl}(\lambda)$. Further, we call p **noticeable** if it is not negligible, and we call p **high** if $p \in 1 - o(1)$. Lastly, we call p **overwhelming** if $1 - p$ is negligible.

Similarly, let $f : \mathbb{N} \to \mathbb{R}$ be some parameter. We call f **polynomial** if $f \in \mathsf{poly}_{\mathbb{R}}(\lambda)$ (not to be confused with algebraic polynomials). Additionally, f is called **exponential** if $f \in 2^{O(\lambda)}$, **subexponential** if $f \in \bigcup_{c \in [0,1)} 2^{O(\lambda^c)}$ and **quasi-polynomial** if $f \in 2^{\mathsf{poly}(\log \lambda)}$. Lastly, f is **linear** if $f \in O(\lambda)$ and **sublinear** if $f \in \bigcup_{c \in [0,1)} O(\lambda^c)$.

Stochastic. In this work, we will only consider discrete distributions.

Let S be a set and $\mathcal{D}$ a distribution over S. If S is finite, we will write $x \leftarrow S$ to denote a random variable x that is sampled uniformly at random from S. We will write $x \leftarrow \mathcal{D}$ to denote that x is distributed according to $\mathcal{D}$. If $s \in S$ is some fixed element, we denote by $\mathcal{D}(s)$ the probability that is assigned by $\mathcal{D}$ to s.

Given two distributions $\mathcal{D}_1, \mathcal{D}_2$ over the set S, we define their **statistical distance** by

$$\Delta(\mathcal{D}_1, \mathcal{D}_2) := \frac{1}{2} \sum_{s \in S} |\mathcal{D}_1(s) - D_2(s)|. \tag{195}$$

Let $n \in \mathbb{N}$, $B \in \mathbb{R}_{>0}$ and let $\mathcal{D}$ be a discrete distribution over $\mathbb{R}^n$. If the support of $\mathcal{D}$ is contained in the ball $\{x \in \mathbb{R}^n \mid B \geq ||x||_\infty\}$, we will call $\mathcal{D}$ **bounded** by B.

Algorithms

I decided to not base the results of this work on any specific computational model. The reason for this is that, while the computational model of Turing machines is very popular in theoretic computer science, we would face some theoretical problems when attempting to implement the attack algorithms and reductions presented in this work by Turing machines. Notoriously, for $n \in \mathbb{N}$ not a power of two, Turing machines are unable to sample a random number $x \leftarrow \{1, \ldots, n\}$ within an a priori fixed number of steps while only having access to random bits. Hence, to give strict time bounds for the algorithms we consider in this work, we will not rely on the computational model of Turing machines. Instead, we will follow a dualistic approach:

1. When considering algorithms of the schemes we are attacking in this work, we will loosely follow the approach of Maurer [Mau02] and not restrict their computational resources. In fact, if $\mathcal{B}$ is a subalgorithm of a scheme we analyze and if we do not mathematically specify how $\mathcal{B}$ operates, then we allow $\mathcal{B}$ to be any *stateless randomized function.* Note that this allows $\mathcal{B}$ to be incomputable and makes counting the resources resp. operations used by $\mathcal{B}$ intractable.

2. On the other side, all adversaries and reductions of this work are described by a list of arithmetic instructions such that they can be implemented by any *arithmetic* computational model that is capable of basic control flow elements and memory management. While we will clearly state the set of arithmetic operations a potential arithmetic computational model needs to be able to perform these algorithms, we will not discuss how if-then-else branching, while loops, memory addressing and other computational tasks are implemented.

Occasionally, we will talk about **PPT** algorithms. By this, we will mean algorithms that can be implemented by binary probabilistic Turing machines with polynomial time bounds. However, the notion of PPT algorithms will not be relevant for our results, we will only use this notion to motivate them.

Arithmetic Operations. Over a ring R, there is a set of arithmetic operations an algorithm presented here may perform in one step. We follow the idea of [AAB15] and consider the following set of operations over R: additions, multiplications, subtractions, divisions (if possible), zero-testing and calling constants as 0 and 1. Further, there are two randomized operations: sampling $r \leftarrow R$

uniformly at random if R is finite and sampling $b \leftarrow \{0, 1\} \subseteq R$ uniformly at random. Since we consider algorithms instead of circuits in this work, we include additional operations as receiving, sending, copying, storing and reading elements of R.

Let us discuss some special rings and their specific operations:

$\mathbb{Z}_2$: The typical boolean operations $\wedge, \vee, \neg$ can be emulated by a constant number of operations over $\mathbb{Z}_2$. Hence, we will call arithmetic operations over $\mathbb{Z}_2$ **bit operations**.

$\mathbb{Z}_q$: The finite rings $\mathbb{Z}_q$ for $q \geq 2$ will be the most important domain for arithmetic operations in this work. In addition to the already introduced arithmetical operations over $\mathbb{Z}_q$, we also allow for a **conversion operation** by which an algorithm may interpret an element $x \in \mathbb{Z}_q$ as an integer $x \in \left\{-\frac{q-1}{2}, \dots, \frac{q-1}{2}\right\}$ and $x \in \left\{-\frac{q}{2}, \dots, \frac{q}{2} - 1\right\}$, respectively. We will call this an arithmetic operation over $\mathbb{Z}_q$ *and* $\mathbb{Z}$. Vice versa, an algorithm can map elements from $\mathbb{Z}$ to $\mathbb{Z}_q$ by computing their residue classes modulo q. We will also count this as an arithmetic operation over $\mathbb{Z}_q$ *and* $\mathbb{Z}$.

Almost all arithmetic operations over $\mathbb{Z}_q$ can be emulated by $O(\log(q)^2)$ bit operations. An exception are sampling uniformly random elements from $\mathbb{Z}_q$, which can only be approximated by an a priori fixed number of bit operations.

$\mathbb{Q}$: We will allow for divisions over $\mathbb{Z}$ and, hence, count arithmetic operations over $\mathbb{Q}$ as arithmetic operations over $\mathbb{Z}$. Over $\mathbb{Z}$ and $\mathbb{Q}$, we additionally allow for comparing elements and computing residues of integers modulo other integers. Further, we allow for **range sampling** over $\mathbb{Z}$. I.e., given numbers $a, b \in \mathbb{Z}$ with $a \leq b$, an algorithm may sample an element $x \leftarrow \{a, \dots, b\}$ uniformly at random as a probabilistic arithmetic operation.

Note that operations over $\mathbb{Z}$ and operations over $\mathbb{Z}_2$ are in general not equivalent from a complexity-theoretic point of view. For example, the algorithm that computes and outputs 2^{2^n} can be implemented by using n operations over $\mathbb{Z}$ (by squaring 2 n-times). However, over $\mathbb{Z}_2$, we would need an exponential number of operations. We can circumvent this problem by bounding all integers that may occur during an operation over $\mathbb{Z}$. If all involved integers are bounded by some B, then the operation can be simulated by $O\big(\log(B^2)\big)$ bit operations.

$\mathbb{R}$: Algorithms over $\mathbb{R}$ are a prime example for algorithms that can not be exactly simulated by Turing machines. When considering operations over $\mathbb{R}$, we allow for comparing values and sampling from integer ranges.

Additionally, whenever we are given a natural extension of rings $R \subset S$ we allow interpreting elements of R as elements of S. We count this as an operation over S.

Algorithms. As explained above, we allow algorithms to be randomized functions, i.e., for each possible input x, the algorithm $\mathcal{A}$ specifies an output distribution $\mathcal{A}(x)$. If for each possible input x, the support of $\mathcal{A}(x)$ consists of only one element, then we call $\mathcal{A}$ a **deterministic** algorithm, otherwise it is **probabilistic** or **randomized**.

Algorithms may **abort** and **terminate** without outputting anything. If a deterministic algorithm $\mathcal{A}$ aborts on input x, we write $\perp \leftarrow \mathcal{A}(x)$ or $\perp = \mathcal{A}(x)$. If a deterministic algorithm $\mathcal{A}$ outputs y on input we write $y \leftarrow \mathcal{A}(x)$ or $y = \mathcal{A}(x)$. If $\mathcal{A}$ is probabilistic, we write $y \leftarrow \mathcal{A}(x)$ to denote that y is a random variable distributed according to the output distribution of $\mathcal{A}$ on input x.

In general, when we informally describe an algorithm $\mathcal{A}$ and this algorithm makes use of a subalgorithm $\mathcal{B}$ or communicates with a party $\mathcal{B}$, we always implicitly assume that $\mathcal{A}$ aborts if $\mathcal{B}$ aborts or behaves in an unexpected way that is not caught by $\mathcal{A}$.

Statefulness and Statelessness. Algorithms may maintain states that can be modelled as implicit inputs and outputs that are given to the algorithm at every call.

A **stateless** algorithm erases resp. resets its state after it halts and outputs something. Note that the output distribution of a randomized stateless algorithm is **memoryless**.

A **stateful** algorithm is allowed to retain its state when it pauses and outputs an element s.t. it can continue its computation after receiving additional input. It is expected to reset its state, however, as soon as it reaches a definitive terminal point. When considering interactive games, we will usually allow the adversaries and challengers to be stateful algorithms that only reset their states at the end of the game.

Time Complexity. If an algorithm is described in this work by a finite list of arithmetic operations and control flow elements, then we define its **time complexity** as the maximum number of arithmetic operations it performs with respect to the number of ring elements of its input. While we count the arithmetic operations of an algorithm over each ring, we will omit the overhead costs for if-then-else forks, while loops and memory management. As far as it concerns this work, this is admissible, since the costs of those flow control tasks are not significant in the algorithms presented here.

When we speak of the time complexity of an algorithm $\mathcal{A}$, we implicitly require that $\mathcal{A}$ has a finite description by arithmetic operations and basic control flow elements.

An algorithm is called **efficient** if it is of polynomial time.

Parameters. All non-constant quantities in this work are parametrized by a **security parameter**, which we will usually denote by λ (in Chapter 1, we will denote the security parameter by n instead of λ because of historical reasons). To avoid information-theoretic problems, we will assume here tacitly that all parameters can be computed by a deterministic efficient algorithm over $\mathbb{Z}_2$ that receives as input 1^λ. When considering runtimes, we will neglect the number of bit operations to compute parameters for a given $\lambda \in \mathbb{N}$.

Chapter 1

Pseudorandom Generators

In Section 1.2, we will discuss the existence of algebraic relations and bounds for their degrees. Afterwards, in Section 1.3, we will turn to the problem of algebraic PRGs and use algebraic relations to derive simple distinguishing attacks for them. Additionally, in Section 1.4, we will investigate the corresponding algebraic search problem and use a search-to-decision reduction to gain a subexponential solving algorithm for it. Finally, in Section 1.5, we will turn our attention to Macaulay matrix-based algorithms and use algebraic relations to reason about bounds for them.

Let us first start with definitions and auxiliary lemmas that are useful for this chapter.

1.1 Preliminaries

1.1.1 Algebraic Preliminaries

Definition 1 (Polynomial Maps)**.** Let k be any field and $n, m \in \mathbb{N}$. We call

$$F : k^n \longrightarrow k^m \tag{1.1}$$

a **polynomial** map, if there are polynomials $f_1, \ldots, f_m \in k[X] = k[X_1, \ldots, X_n]$ s.t. the j-th output of $F(x)$ equals $f_j(x)$ for all $j \in [m]$ and $x \in k^n$.

We define the **degree** of F by

$$\deg F := \max_{j \in [m]} \deg f_j, \tag{1.2}$$

where we tacitly assume that the polynomials $f_1, \ldots, f_m$ are reduced modulo the field equations of k.

Note that each polynomial map is continuous in the Zariski topology.

Definition 2 (Dual Morphisms)**.** Let $F : k^n \to k^m$ be a polynomial map consisting of polynomials $f_1, \ldots, f_m$. The **dual morphism** of F is given by the following morphism of k-algebras

$$\phi : k[Y] \longrightarrow k[X] \tag{1.3}$$

$$Y_i \longrightarrow f_i(X). \tag{1.4}$$

I.e., ϕ maps each polynomial $h(Y_1, \ldots, Y_m) \in k[Y]$ to the polynomial $h(f_1(X), \ldots, f_m(X))$ in $k[X]$ by substituting each occurrence of Y_j in h by $f_j(X_1, \ldots, X_n)$ for each $j \in [m]$.

Definition 3 (Algebraic Independence)**.** We call $f_1, \ldots, f_m$ **algebraically independent** (or **transcendent**) over k if their dual morphism ϕ from Definition 2 is injective.

If ϕ is not injective, we call a non-zero element $h \in \ker \phi$ of its kernel an **algebraic relation** of the elements $f_1, \ldots, f_m$.

The following is a well-known fact from algebra that informally states that at most n elements of $k[X_1, \ldots X_n]$ can be transcendent over k.

Lemma 1. *Let k be any field and let $f_1, \ldots, f_m \in k[X_1, \ldots, X_n]$.*

If $m > n$, then $f_1, \ldots, f_m$ must be algebraically dependent and there must exist an algebraic relation among them.

We will discuss the implications of Lemma 1 in more detail in Section 1.2. Now, let us consider a very important tool for the adversaries of this chapter:

Lemma 2 (Schwartz-Zippel [DL78; Zip79; Sch80])**.** *Let k be any field and let $S_1, \ldots, S_m \subset k$ be finite sets that share all the same cardinality. Let $h \in k[Y_1, \ldots, Y_m]$ be a non-zero polynomial in m variables. We have*

$$\Pr_{y \leftarrow S_1 \times \ldots \times S_m} [h(y) = 0] \leq \frac{\deg h}{\# S_1}. \tag{1.5}$$

Given the importance of the Schwartz-Zippel lemma, we sketch here its proof.

Proof Sketch. Let $h \in k[Y]$ be non-zero. If $m = 1$, it is easy to see that the claim does hold, since h can have at most $\deg h$ distinct roots.

Now, let $m > 1$ and rewrite h as

$$h(Y) = \sum_{i=0}^{d} c_i(Y_1, \ldots, Y_{m-1}) \cdot Y_m^i \tag{1.6}$$

for polynomials $c_0, \ldots, c_d \in k[Y_1, \ldots, Y_{m-1}]$. Assume that c_d is non-zero and note that we have

$$\deg c_i \leq \deg h - i. \tag{1.7}$$

By an inductive argument, for $(y_1, \ldots, y_m) \leftarrow S_1 \times \ldots \times S_m$, the probability that $c_d(y_1, \ldots, y_{m-1})$ is zero is upper bounded by

$$\frac{\deg c_d}{\# S_1} \leq \frac{\deg h - d}{\# S_1}. \tag{1.8}$$

If $c_d(y_1, \ldots, y_{m-1})$ is not zero, then $h(y_1, \ldots, y_{m-1}, Y_m)$ is a non-zero univariate polynomial of degree d. The probability that $h(y_1, \ldots, y_{m-1}, Y_m)$ vanishes on $y_m \leftarrow S_m$ is lower bounded by $\frac{d}{\# S_d}$. The claim follows now by a union bound. □

Finally, let us introduce the notion of algebraic varieties:

Definition 4. Let $I \subset k[X_1, \dots, X_n]$ be an ideal for some field k. We define the **variety** of I (over k) as the following geometric set

$$V(I) := \{x \in k^n \mid \forall f \in I : \ f(x) = 0\}. \tag{1.9}$$

Note that we do not define our varieties over algebraically closed fields and do not require them to be irreducible.

Field Equations

Definition 5. Let $q, n \in \mathbb{N}$ and let k be any field. We define I_q to be the ideal generated by the polynomials $X_1^q - X_1, \dots, X_n^q - X_n$, i.e.

$$I_q = (X_1^q - X_1, \dots, X_n^q - X_n) \subset k[X_1, \dots, X_n]. \tag{1.10}$$

Proposition 3. *If $q = \#k$, then I_q is generated by the field equations of k, and we have*

$$X_i^q - X_i = \prod_{x \in k} (X_i - x). \tag{1.11}$$

Definition 6. Let $q, n, d \in \mathbb{N}$ and let k be any field. By abuse of notation, we set

$$\begin{aligned} k[X]/I_q :=& k[X_1, \dots, X_n]/(X_1^q - X_1, \dots, X_n^q - X_n), \\ k[X]^{\leq d}/I_q :=& k[X_1, \dots, X_n]^{\leq d}/(k[X_1, \dots, X_n]^{\leq d} \cap (X_1^q - X_1, \dots, X_n^q - X_n)). \end{aligned}$$

We define the **degree** of $f \in k[X]/I_q$ by

$$\deg f := \begin{cases} 0, & \text{if } f = 0, \\ \min\left\{d \in \mathbb{N}_0 \mid f \in k[X]^{\leq d}/I_q\right\}, & \text{if } f \neq 0. \end{cases} \tag{1.12}$$

Note that the ring $k[X]/I_q$ is not graded, in contrast to $k[X]$. However, the collection $(k[X]^{\leq d}/I_q)_d$ gives us a **filtration** of $k[X]/I_q$ and allows us to sort elements according to their degree. Analogously to the non-reduced case, we can bound the vector space dimensions of the spaces $k[X]^{\leq d}/I_q$.

Lemma 4. *Let k be any field and $n, d \in \mathbb{N}$. For $n \geq d$, we have*

$$\dim_k k[X]^{\leq d}/I_2 = \binom{n}{d}. \tag{1.13}$$

For $q \geq 2$ and arbitrary $n, d \in \mathbb{N}$ we have

$$\binom{n}{d} \leq \dim_k k[X]^{\leq d}/I_q = \binom{n+d}{d}. \tag{1.14}$$

Proof. Let $n \geq d$. I_2 is generated by the polynomials $X_i^2 - X_i$ for $i \in [n]$. Hence, $k[X]^{\leq d}/I_2$ is generated by all monomials of degree $\leq d$ that contain each variable at most once. These monomials form a basis of $k[X]^{\leq d}/I_2$ and their count is $\binom{n}{d}$.

For general $q, n, d \in \mathbb{N}$, we note that the ideal I_2 contains the ideal I_q. Hence, we get the following chain of surjective linear maps

$$k[X]^{\leq d} \longrightarrow k[X]^{\leq d}/I_q \longrightarrow k[X]^{\leq d}/I_2. \tag{1.15}$$

Since we know the dimensions of $k[X]^{\leq d}$ and $k[X]^{\leq d}/I_2$, the claimed inequalities for the dimension of $k[X]^{\leq d}/I_q$ follow. □

We will now show a lemma that can be seen as a poor man's version of the Schwartz-Zippel Lemma 2. It bounds the number of roots of a multivariate polynomial h even if the degree of h exceeds the size of its field.

Lemma 5. *Let k be a field of finite size q and let $h \in k[Y_1, \dots, Y_m]/I_q$ be non-zero. Then, we have*

$$\Pr_{y \leftarrow k^m}[h(y) = 0] \leq 1 - q^{-\deg h}. \tag{1.16}$$

Proof. Since h is non-zero modulo $I_q = (Y_1^q - Y_1, \dots, Y_m^q - Y_m)$, we can interpret it as a non-zero polynomial in $k[Y]$ that has degree at most $q-1$ in each variable Y_i. Set $d := \deg h$. We will show that we have

$$\#\{y \in k^m \mid h(y) = 0\} \leq q^m - q^{m-d}. \tag{1.17}$$

First assume that no linear polynomial of the form $Y_m - c$ for $c \in k$ divides h (over $k[Y]$). In that case, h can be written as

$$h(Y_1, \dots, Y_m) = \sum_{i \in k} \frac{Y_m^q - Y_m}{Y_m - i} \cdot g_i(Y_1, \dots, Y_{m-1}) \tag{1.18}$$

where each $g_i \in k[Y_1, \dots, Y_m]$ is reduced modulo I_q, non-zero and of degree $\leq d$ (in fact, g_i is a scalar multiple of $h(Y_1, \dots, Y_{m-1}, i)$). Then, we have

$$\#\{y \in k^m \mid h(y) = 0\} \tag{1.19}$$

$$= \sum_{i \in k} \#\{y \in k^{m-1} \mid g_i(y) = 0\}. \tag{1.20}$$

By an inductive argument, the claim now follows.

On the other hand, assume that h is divisible by a linear term $Y_m - c$. Set

$$S := \{c \in k \mid (Y_m - c) | h\}. \tag{1.21}$$

Without loss of generality, we can assume that, for each $c \in S$, $Y_m - c$ divides h only once. Now, note that for each $c \notin S$ the polynomial

$$h(Y_1, \dots, Y_{m-1}, c) \tag{1.22}$$

is non-zero modulo I_q and of degree $< d$. We now have

$$\{y \in k^m \mid h(y) = 0\} \tag{1.23}$$

$$= \big(k^{m-1} \times S\big) \cup \bigcup_{c \in k \setminus S} \big(\{y \in k^{m-1} \mid h(y,c) = 0\} \times \{c\}\big). \tag{1.24}$$

By an inductive argument, it follows

$$\#\{y \in k^m \mid h(y) = 0\} \leq q^{m-1} \cdot \#S + (q - \#S) \cdot (q^{m-1} - q^{m-1-(d-1)}) \tag{1.25}$$

$$= q^m - (q - \#S) \cdot q^{m-d} \leq q^m - q^{m-d}, \tag{1.26}$$

since $\#S < q$. This finishes the proof of the lemma. □

Field Extensions

Definition 7. Let $k, \overline{k}$ be fields. If there exists a homomorphism of rings $\iota : k \to \overline{k}$, we will call the pair $k, \overline{k}$ a **field extension**. Note that each ring homomorphism must send 1 to 1, therefore each ring homomorphism must be injective on fields. In particular, $\iota : k \to \overline{k}$ is one-to-one and—without loss of generality—we can assume that k is a subset of $\overline{k}$. By abuse of notation, we will denote field extensions always as subset-relationships $k \subset \overline{k}$.

We define the **degree** of the field extension $k \subseteq \overline{k}$ by

$$[\overline{k} : k] := \dim_k(\overline{k}). \tag{1.27}$$

The following is a well-known fact from algebra and the theory of field extensions.

Lemma 6. *Let $k \subset \overline{k}$ be an extension of finite fields. Then, $k \subset \overline{k}$ is* simple*, i.e., there exists an element $\zeta \in \overline{k}$ s.t. $\overline{k} = k[\zeta]$. Concretely, each element of $\overline{k}$ can be written as $f(\zeta)$ where $f \in k[Z]$ is a univariate polynomial.*

Note that each simple and finite field extension $k \subset \overline{k}$ that is generated by one element ζ can be written as

$$k[\zeta] = \overline{k} \cong k \oplus \zeta \cdot k \oplus \ldots \oplus \zeta^{r-1} \cdot k \cong k^r \tag{1.28}$$

where $r = [\overline{k} : k]$. I.e., as a k-vector space $\overline{k}$ has the basis $1, \ldots, \zeta^{r-1}$.

Proposition 7. *The map*

$$\psi : k^r \longrightarrow \overline{k} \tag{1.29}$$

$$(b_1, \ldots, b_r) \longmapsto b_1 + b_2 \cdot \zeta + \ldots + b_r \cdot \zeta^{r-1} \tag{1.30}$$

is an isomorphism of k-vector spaces.

Now, let us turn to the computational aspects of field extensions. First, we want to determine how we can emulate operations over an extension field $\overline{k}$ by arithmetic operations over its base field k. Since the size of $\overline{k}$ is finite, it is—up to isomorphism—determined by its cardinality. Let $r = [\overline{k} : k]$ be the degree of the extension, then $\overline{k}$ is isomorphic to

$$\overline{k} \cong k[Z]/(s(Z)) \tag{1.31}$$

where $s(Z)$ is a univariate irreducible polynomial over k of degree r. Addition of elements of $k[Z]/(s(Z))$ can be computed entry-wise, while for the product of two elements with representatives $a, b \in k[Z]^{\leq r-1}$ one first computes the polynomial

$$a \cdot b = (a_0 + a_1 Z + \ldots + a_{r-1} Z^{r-1}) \cdot (b_0 + b_1 Z + \ldots + b_{r-1} Z^{r-1}) \tag{1.32}$$

$$= \sum_{i,j=0}^{r-1} a_i b_j Z^{i+j} = \sum_{\ell=0}^{2r-2} Z^\ell \cdot \sum_{i=\max(0,\ell-r+1)}^{\ell} a_i b_{\ell-i} \tag{1.33}$$

and then reduces the higher monomials $Z^r, \ldots, Z^{2r-2}$ modulo $s(Z)$. Inversion of a non-zero element with representation $a \in k[Z]^{\leq r-1}$ can be performed by using the extended Euclidean algorithm to find $b, t \in k[Z]^{\leq r-1}$ s.t.

$$a \cdot b + s \cdot t = 1. \tag{1.34}$$

The multiplicative inverse of $a + (s(Z))$ is then given by $b + (s(Z))$. The time complexity for this is given by $O(r^2)$ arithmetic operations over k. Hence, we can conclude the following:

Proposition 8. *Let $k \subset \overline{k}$ be a field extension of degree $r = [\overline{k} : k]$. Given a representation $k[Z]/(s(Z))$ of $\overline{k}$, each arithmetic operation of $\overline{k}$ can be simulated by $O(r^2)$ arithmetic operations over k.*

Finally, given an integer r and a field k of *finite* size q, it is left to compute an extension of k of degree r by a representation $k[Z]/(s(Z))$, i.e., by an irreducible univariate polynomial $s \in k[Z]$ of proper degree r. A simple algorithm for this task is given by a sieving method where we enumerate all polynomials h of degree $< r$ and mark all multiples of h of degree r as reducible. Such an approach would need $O(q^{2r})$ arithmetic operations over k. A better algorithm is given by Rabin [Rab80] who observed that a random polynomial of proper degree r is irreducible with probability $\approx 1/r$. Together with a fast test for irreducibility that utilizes $O(\log(r)^3 \cdot r^2 \cdot \log(q))$ operations over k, this yields an algorithm for finding irreducible polynomials of proper degree r that needs $O(\log(r)^3 \cdot r^3 \cdot \log(q) \cdot \lambda)$ arithmetic operations over k. However, since this algorithm is based on rejection-sampling, it has a small probability of failure. The first efficient deterministic algorithm for finding an irreducible polynomial has been given by Shoup [Sho88]. His algorithm is based on advanced knowledge of algebraic number theory.

Lemma 9 ([Sho88])**.** *Let k be a field of characteristic $p > 0$ and size $q = p^e$.*

There is an algorithm that on input $r \in \mathbb{N}$ outputs an irreducible univariate polynomial $s \in k[Z]$ of proper degree r by utilizing

$$O(\log(p) \cdot e^{2.1} \cdot r^{4.1} + \log(p)^2 \cdot r^{4.1} + \log(p)^3 \cdot p^{1/2} \cdot r^{3.1}) \subset O(q \cdot r^{4.1}) \quad (1.35)$$

arithmetic operations over $\mathbb{Z}_p$.

To keep our algorithms simple and deterministic, we will use in this work Shoup's algorithm when we need to compute field extensions of certain degrees. However, note that in reality one might prefer Rabin's algorithm, since it is faster. In particular, when p is of exponential size and r is polynomial, Rabin's algorithm has a polynomial time complexity, while Shoup's algorithm needs an exponential number of operations.

1.1.2 Cryptographic Preliminaries

We will first give a very general definition for pseudorandom generators together with a security game for their pseudorandomness. Afterwards, we will specify the algebraic pseudorandom generators that we will study in this work.

Definition 8 (Pseudorandom Number Generators)**.** A **pseudorandom number generator** (PRG) is a triple of three families $\mathcal{F} = (\mathcal{F}_\lambda)_\lambda$, $\mathcal{X} = (\mathcal{X}_\lambda)_\lambda$, $\mathcal{Y} = (\mathcal{Y}_\lambda)_\lambda$ of discrete distributions. The distributions have the following syntax:

$\mathcal{X}$: Each $\mathcal{X}_\lambda$ is a distribution of *seeds* for the PRG.

$\mathcal{Y}$: Each $\mathcal{Y}_\lambda$ is a distribution of *true random values* for the PRG.

$\mathcal{F}$: Each $\mathcal{F}_\lambda$ is a distribution of deterministic functions that map elements of the support of $\mathcal{X}_\lambda$ to elements of the support of $\mathcal{Y}_\lambda$.

We will denote the PRG with the distributions $\mathcal{F}$, $\mathcal{X}$ and $\mathcal{Y}$ by

$$\mathcal{F} : \mathcal{X} \longrightarrow \mathcal{Y}. \tag{1.36}$$

If each $\mathcal{X}_\lambda$ is the uniform distribution over some finite set X_λ and if each $\mathcal{Y}_\lambda$ is the uniform distribution over some finite set Y_λ, we will denote the PRG $\mathcal{F}$, $\mathcal{X}$, $\mathcal{Y}$ simply by

$$\mathcal{F} : X \longrightarrow Y, \tag{1.37}$$

where $X = (X_\lambda)_\lambda$ and $(Y_\lambda)_\lambda$ are the families of the corresponding sets.

We will define the pseudorandomness of PRGs by a game between an adversary and a challenge. The game consists of an *offline* and an *online* phase: in the offline or preprocessing phase, the adversary sees a deterministic function $F \leftarrow \mathcal{F}$ and may compute a hint for itself. In the online phase, the adversary sees the deterministic function F together with a potential output value y that is either truly random, i.e., $y \leftarrow \mathcal{Y}$, or pseudorandom, i.e., $y = F(x)$ for $x \leftarrow \mathcal{X}$. In the online phase, the adversary may make use of the hint it computed in the offline phase. However, besides the hint, it may retain no memory of the offline phase.

Game 2 (Security Game for PRGs)**.** Let $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$ be a PRG. We define the **pseudorandomness game** of $\mathcal{F}$ as the following game between a stateful challenger $\mathcal{C}$ and an adversary $\mathcal{A} = (\mathcal{A}_{\mathsf{off}}, \mathcal{A}_{\mathsf{on}})$ that consists of a stateless offline algorithm $\mathcal{A}_{\mathsf{off}}$ and a stateless online algorithm $\mathcal{A}_{\mathsf{on}}$. The game is parametrized by λ.

Phase 1: $\mathcal{C}$ samples a deterministic function $F \leftarrow \mathcal{F}_\lambda$ and sends it to $\mathcal{A}_{\mathsf{off}}$.

Phase 2: $\mathcal{A}_{\mathsf{off}}$ computes a hint ht and sends it to $\mathcal{C}$.

Phase 3: $\mathcal{C}$ draws a bit $b \leftarrow \{0,1\}$. If $b = 0$, it samples a preimage $x \leftarrow \mathcal{X}_\lambda$ and sets $y := F(x)$. If $b = 1$, it samples $y \leftarrow \mathcal{Y}_\lambda$.

Phase 4: $\mathcal{C}$ sends the tuple (F, y, ht) to $\mathcal{A}_{\mathsf{on}}$.

Phase 5: $\mathcal{A}_{\mathsf{on}}$ receives (F, y, ht) and must decide which bit b has been drawn by $\mathcal{C}$. It makes some computations on its own without interacting with $\mathcal{C}$ and finally sends a bit b' to $\mathcal{C}$.

$\mathcal{A}$ wins an instance of this game iff $b = b'$ holds at the end.

We define $\mathcal{A}$'s **offline time complexity** as the number of arithmetic operations performed by $\mathcal{A}_{\mathsf{off}}$ and its **online time complexity** as the number of arithmetic operations[1] performed by $\mathcal{A}_{\mathsf{on}}$. The **(total) time complexity** of $\mathcal{A}$ is the sum of its online and offline time complexity.

Further, for a value y in the support of $\mathcal{Y}_\lambda$ and a function F in the support of $\mathcal{F}_\lambda$, we set

$$\mathcal{A}(F, y) := \mathcal{A}_{\mathsf{on}}(F, y, \mathcal{A}_{\mathsf{off}}(F)). \tag{1.38}$$

[1] Note that $\mathcal{A}_{\mathsf{on}}$ needs to read the hint ht of $\mathcal{A}_{\mathsf{off}}$, hence the time complexity of $\mathcal{A}$ is at least as large as the size of ht.

The reason why we split the pseudorandomness Game 2 into an offline and an online phase is because it is usual in cryptographic literature to assume that $\mathcal{F}_\lambda$ always outputs the same fixed function. I.e., a lot of cryptographic works assume the existence of a fixed series $(F_n)_n$ of deterministic functions s.t. the output of each $F_n : \{0,1\}^n \to \{0,1\}^m$ is indistinguishable from a random bit string. Against this type of fixed PRGs, *preprocessing* attacks are admissible in which the offline part may take an exponentially large amount of time to compute a small hint for F_n that reduces the online time of the attack. This motivates to consider offline and online time complexity separately.

Note that we did not require in Definition 9 the distributions $\mathcal{F}_\lambda$, $\mathcal{X}_\lambda$ and $\mathcal{Y}_\lambda$ to be efficiently samplable or even computable. If $\mathcal{F}_\lambda, \mathcal{X}_\lambda$ or $\mathcal{Y}_\lambda$ can not be efficiently sampled, then $\mathcal{C}$ in Game 2 cannot be efficiently simulated. This is not a problem in this work, since the PRGs that we study here will always have efficiently samplable distributions.

Definition 9 (Pseudorandomness)**.** Let $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$ be a PRG and let $\mathcal{A}$ be an adversary for Game 2.

We define $\mathcal{A}$'s **advantage**[2] against the pseudorandomness of $\mathcal{F}$ by

$$\begin{aligned} &\mathsf{adv}^{\mathsf{PRG}}_{\mathcal{F}}(\mathcal{A}) := 2 \cdot \Pr[\mathcal{A} \text{ wins}] - 1 && (1.39)\\ &= \Pr_{\substack{F \leftarrow \mathcal{F}_\lambda \\ x \leftarrow \mathcal{X}_\lambda}}[\mathcal{A}(F, F(x)) = 0] + \Pr_{\substack{F \leftarrow \mathcal{F}_\lambda \\ y \leftarrow \mathcal{Y}_\lambda}}[\mathcal{A}(F, y) = 1] - 1 && (1.40)\end{aligned}$$

where we take the probability over the randomness of $\mathcal{A}$ and $\mathcal{C}$. Note that $\mathsf{adv}^{\mathsf{PRG}}_{\mathcal{F}}(\mathcal{A})$ is a function in λ.

We say that $\mathcal{F}$ is **pseudorandom against a class A** of adversaries if the advantage of each adversary $\mathcal{A} \in \mathbf{A}$ in Game 2 is negligible, i.e., $\mathsf{adv}^{\mathsf{PRG}}_{\mathcal{F}}(\mathcal{A}) \in \mathsf{negl}(\lambda)$. We will call $\mathcal{F}$ **(uniformly) pseudorandom** if $\mathcal{F}$ is pseudorandom against the class of algorithms $(\mathcal{A}_{\mathsf{off}}, \mathcal{A}_{\mathsf{on}})$ where $\mathcal{A}_{\mathsf{off}}$ and $\mathcal{A}_{\mathsf{on}}$ are PPT. Finally, we will call $\mathcal{F}$ **non-uniformly pseudorandom** if $\mathcal{F}$ is pseudorandom against the class of algorithms $(\mathcal{A}_{\mathsf{off}}, \mathcal{A}_{\mathsf{on}})$ where only $\mathcal{A}_{\mathsf{on}}$ is PPT.

We will now define the class of PRGs that is of interest for us:

Definition 10 (Algebraic Pseudorandom Generators)**.** Let $k = (k_\lambda)_\lambda$ be a family of fields and let $n, m, d \in \mathbb{N}$ be parameters $n = n(\lambda), m = m(\lambda), d = d(\lambda)$. Further, let $S = (S_\lambda)_\lambda$ and $T = (T_\lambda)_\lambda$ be families of finite subsets of k, i.e., we have for each $\lambda \in \mathbb{N}$

$$S_\lambda, T_\lambda \subseteq k_\lambda. \tag{1.41}$$

We call a PRG $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$ a **PRG of degree** d **(over** k**)** (or an **algebraic PRG**), if the following requirements are met:

1. For each $\lambda \in \mathbb{N}$, $\mathcal{X}_\lambda$ is the uniform distribution over the finite set $S_\lambda^{n(\lambda)} \subset k_\lambda^{n(\lambda)}$.

[2] Note that we do not use the absolute value of $2 \cdot \Pr[\mathcal{A} \text{ wins}] - 1$ to define the advantage of $\mathcal{A}$ against $\mathcal{F}$. This is because we want to quantify our security definition over all possible adversaries of a certain class. Now, an adversary in the above game can trivially lose by aborting. Such an adversary would have an advantage of -1, which may appear unconventional. However, the upside of this definition is that we can allow for adversaries that abort the above game in some cases.

2. For each $\lambda \in \mathbb{N}$, $\mathcal{Y}_\lambda$ is the uniform distribution over the finite set $T_\lambda^{m(\lambda)} \subset k_\lambda^{m(\lambda)}$.
3. For each $\lambda \in \mathbb{N}$, $\mathcal{F}_\lambda$ is a distribution of polynomial maps $F : k_\lambda^{n(\lambda)} \to k_\lambda^{m(\lambda)}$ of degree d over k_λ.

In this case, we will denote the PRG by $\mathcal{F} : S^n \to T^m$ instead of $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$.

When dealing with algebraic PRGs, we will introduce some conventions to ease notation:

Convention 1. With regard to the cryptographic literature about PRGs, we will always assume that the seed size of an algebraic PRG $\mathcal{F} : S^n \to T^m$ equals the security parameter λ, i.e.

$$n(\lambda) = \lambda. \tag{1.42}$$

In this case, we will call $m = m(n)$ the **stretch** of $\mathcal{F}$.

Further, when considering adversaries against the pseudorandomness of PRGs $\mathcal{F}$ of *constant* degree d, we will always assume that the challenger in Game 2 hands the adversary a description of $F \leftarrow \mathcal{F}_n$ that consists of $m(n)$ polynomials in $k_n[X_1, \ldots, X_n]$ of degree d. We expect that these polynomials are expressed by $m(n) \cdot \binom{n+d}{d}$ elements of k_n.

Finally, in the context of algebraic PRGs, we will—by abuse of notation—speak of the field k when we actually mean the family of fields $(k_n)_n$, and for a fixed $n \in \mathbb{N}$, we will speak of k when we actually mean k_n.

Definition 11. Let $\mathcal{F} : S^n \to T^m$ be a PRG of degree d. We will say that $\mathcal{F}$ is of **poly-stretch** if there is a constant $e \in \mathbb{R}_{>0}$ s.t. we have for all $n \in \mathbb{N}$

$$m(n) \geq n^{1+e}. \tag{1.43}$$

1.2 Algebraic Relations

Let $f_1, \ldots, f_m \in k[X_1, \ldots, X_n]$ be a collection of $m > n$ polynomials of some degree d. Because of Lemma 1, we know that there must exist an algebraic relationship among $f_1, \ldots, f_m$, i.e., a polynomial $h \in k[Y_1, \ldots, Y_m]$ s.t.

$$h \neq 0, \tag{1.44}$$

$$h(f_1, \ldots, f_m) = 0 \in k[X]. \tag{1.45}$$

While pure algebra tells us that such a polynomial h must exist, it does not give us an upper bound for the degree of h or an algorithm to find h. We will address both problems in this section.

Let us first bound the degree of h.

Theorem 10. *Let k be a field and let $d \in \mathbb{N}$ be constant. Further, let $m : \mathbb{N} \to \mathbb{N}$ be a function with*

$$m(n) \geq 2^{2d-1} \cdot d^{d-1} \cdot n. \tag{1.46}$$

Define the function $D : \mathbb{N} \to \mathbb{N}$ as follows

$$D(n) := \min\left\{ L \in \mathbb{N} \;\middle|\; \binom{m(n)+L}{L} > \binom{n+d \cdot L}{d \cdot L} \right\}. \tag{1.47}$$

Then, the following claims hold:

1. *For $n \geq 2d$ and $d > 1$, we have*

$$D(n) \leq \left\lceil \left(\frac{(2 \cdot n)^d}{m(n)} \right)^{1/(d-1)} \right\rceil. \tag{1.48}$$

2. *For all $n \in \mathbb{N}$ and $d > 1$, we have*

$$D(n) \geq \frac{1}{d} \cdot \left(\frac{n^d}{2 \cdot m(n)} \right)^{1/(d-1)}. \tag{1.49}$$

3. *For each $n \in \mathbb{N}$ and each collection $f_1, \ldots, f_m \in k[X]^{\leq d}$ of $m(n)$ functions of degree d over n variables there exists a polynomial $h \in k[Y]$ with:*

$$h \neq 0, \tag{1.50}$$

$$\deg h \leq D(n), \tag{1.51}$$

$$h(f_1, \ldots, f_m) = 0 \in k[X]. \tag{1.52}$$

Note that the first and second part of Theorem 10 imply together

$$D(n) \in \Theta\Big(\big(n^d/m\big)^{1/(d-1)} \Big). \tag{1.53}$$

Corollary 11. *In the situation of Theorem 10, if $m \geq n^{1+e}$ for some constant $e > 0$, we have*

$$D(n) \in 2^{d/(d-1)} \cdot n^{1-e/(d-1)} + O(1). \tag{1.54}$$

The first claim of Theorem 10 has been shown in [Üna23b]:

Lemma 12. *Let $d \in \mathbb{N}$ and let $m : \mathbb{N} \to \mathbb{N}$ s.t. $m(n) \geq 2^{2d-1} \cdot d^{d-1} \cdot n$ for all n. Then, we have for all integers $n \geq 2d$*

$$\binom{m(n) + L(n)}{L(n)} > \binom{n + d \cdot L(n)}{d \cdot L(n)} \tag{1.55}$$

where $L(n) = \left\lceil \left(\frac{(2n)^d}{m} \right)^{\frac{1}{d-1}} \right\rceil$.

Proof. Let $n \geq 2d$. Then, we have $2n \geq n + dL$, since

$$n \geq dL \iff n \geq d \cdot \left\lceil \left(\frac{(2n)^d}{m} \right)^{\frac{1}{d-1}} \right\rceil \tag{1.56}$$

$$\Leftarrow \quad n \geq d \cdot \left(\left(\frac{(2n)^d}{m} \right)^{\frac{1}{d-1}} + 1 \right) \tag{1.57}$$

$$\iff n - d \geq d \cdot \left(\frac{(2n)^d}{m} \right)^{\frac{1}{d-1}} \tag{1.58}$$

$$\iff (n-d)^{d-1} \geq d^{d-1} \cdot \frac{(2n)^d}{m} \tag{1.59}$$

$$\iff m \geq d^{d-1} \cdot 2^d \cdot \left(\frac{n}{n-d} \right)^{d-1} \cdot n \tag{1.60}$$

$$\overset{n \geq 2d}{\Leftarrow} \quad m \geq d^{d-1} \cdot 2^d \cdot 2^{d-1} \cdot n \tag{1.61}$$

where the last inequality is required in the premise of the lemma.

Now, for the claimed inequality of the lemma, we have the following chain of equivalent inequalities

$$\binom{m+L}{L} > \binom{n+dL}{dL} \tag{1.62}$$

$$\iff \frac{(m+L)\cdots(m+1)}{L!} > \frac{(n+dL)\cdots(n+1)}{(dL)!} \tag{1.63}$$

$$\iff (m+L)\cdots(m+1)\cdot(dL)\cdots(L+1) > (n+dL)\cdots(n+1) \tag{1.64}$$

Note that we have for all $n \in \mathbb{N}$ the inequalities

$$(m+L)\cdots(m+1) > m^L, \tag{1.65}$$

$$(dL)\cdots(L+1) > L^{(d-1)L}. \tag{1.66}$$

For the right-hand side, we have

$$(n+dL)\cdots(n+1) \leq (n+dL)^{dL} \leq (2n)^{dL} = 2^{dL}\cdot n^{dL}. \tag{1.67}$$

By using the inequalities Eqs. (1.65) to (1.67), we see that Eq. (1.64) is implied by the inequality

$$m^L \cdot L^{(d-1)L} \geq 2^{dL}\cdot n^{dL}. \tag{1.68}$$

By reducing Eq. (1.68) to the L-th root, we get the equivalent inequality

$$m\cdot L^{d-1} \geq 2^d \cdot n^d. \tag{1.69}$$

Eq. (1.69) holds since we have $L \geq \left(\frac{(2n)^d}{m}\right)^{\frac{1}{d-1}}$. □

For the second point of Theorem 10, let us show the following lemma:

Lemma 13. *Let $n, m, d, D \in \mathbb{N}$ with $d > 1$ s.t.*

$$m \geq 2^{2d-1}\cdot d^{d-1}\cdot n, \tag{1.70}$$

$$\binom{m+L}{L} > \binom{n+d\cdot L}{d\cdot L}. \tag{1.71}$$

Then, we have

$$L \geq \frac{1}{d}\cdot\left(\frac{n^d}{2m}\right)^{1/(d-1)}. \tag{1.72}$$

Proof. Let us assume—for the sake of contradiction—that we have

$$L < \frac{1}{d}\cdot\left(\frac{n^d}{2m}\right)^{1/(d-1)}. \tag{1.73}$$

We will first show the following inequality for m and L

$$2m \geq m+L. \tag{1.74}$$

Indeed, we have

$$L < \frac{1}{d} \cdot \left(\frac{n^d}{2m}\right)^{1/(d-1)} \leq \frac{1}{d} \cdot \left(\frac{n^d}{2^{2d} \cdot d^{d-1} \cdot n}\right)^{1/(d-1)} \leq \frac{1}{d} \cdot n \leq m. \tag{1.75}$$

As before, the inequality

$$\binom{m+L}{L} > \binom{n+d \cdot L}{d \cdot L} \tag{1.76}$$

is equivalent to

$$(dL) \cdots (L+1) > \frac{(n+dL) \cdots (n+1)}{(m+L) \cdots (m+1)}. \tag{1.77}$$

Therefore, we have

$$\begin{aligned} (dL)^{(d-1)L} \geq & (dL) \cdots (L+1) && (1.78) \\ \geq & \frac{(n+dL) \cdots (n+1)}{(m+L) \cdots (m+1)} && (1.79) \\ \geq & \frac{n^{dL}}{(m+L)^L} && (1.80) \\ \geq & \frac{n^{dL}}{(2m)^L} = \left(\frac{n^d}{2m}\right)^L. && (1.81) \end{aligned}$$

We reduce both sides to the $(d-1)L$-th root to see

$$dL \geq \left(\frac{n^d}{2m}\right)^{1/(d-1)}, \tag{1.82}$$

which is equivalent to $L \geq \frac{1}{d} \cdot \left(n^d/(2m)\right)^{1/(d-1)}$. Hence, we reach a contradiction and the deduced inequality must hold whenever the premises of our lemma are fulfilled. □

The last point of Theorem 10 is implied by the following lemma:

Lemma 14. *Let $F : k^n \to k^m$ be a polynomial map of degree d that is computed by polynomials $f_1, \ldots, f_m \in k[X]$. Denote by*

$$\begin{aligned} \phi : k[Y_1, \ldots, Y_m] &\longrightarrow k[X_1, \ldots, X_n] && (1.83) \\ Y_j &\longmapsto f_j(X) && (1.84) \end{aligned}$$

the dual morphism of F.

If we have

$$\binom{m+L}{L} > \binom{n+dL}{dL} \tag{1.85}$$

for some $L \in \mathbb{N}$, then $\ker \phi$ must contain a non-zero element of degree L.

Proof. Let $k[Y]^{\leq L}$ be the space of polynomials of $k[Y]$ of degree $\leq L$, and let $k[X]^{\leq dL}$ be the space of polynomials of $k[X]$ of degree $\leq dL$.

We claim that ϕ maps $k[Y]^{\leq L}$ to $k[X]^{\leq dL}$. Indeed, we have for each $h \in k[Y]$

$$\deg \phi(h) = \deg h(f_1(X), \ldots, f_m(X)) \leq d \cdot \deg h. \tag{1.86}$$

This is, because ϕ replaces each monomial $Y_{i_1} \cdots Y_{i_\ell}$ of h by $f_{i_1}(X) \cdots f_{i_\ell}(X)$. The degree of this product is bounded by

$$\deg f_{i_1} + \ldots + \deg f_{i_\ell} \leq \ell \cdot d. \tag{1.87}$$

Hence, we can restrict ϕ to the following map

$$\phi^{\leq L} : k[Y]^{\leq L} \longrightarrow k[X]^{\leq dL} \tag{1.88}$$

$$h \longmapsto \phi(h). \tag{1.89}$$

While $\phi^{\leq L}$ is not a ring morphism any more, it is still a linear map of vector spaces over k. The dimension formula for linear maps now implies

$$\dim_k \ker \phi^{\leq L} \geq \dim_k k[Y]^{\leq L} - \dim_k k[X]^{\leq dL} = \binom{m+L}{L} - \binom{n+dL}{dL}.$$

According to the premise of the lemma, $\binom{m+L}{L} - \binom{n+dL}{dL}$ is greater than 0. Hence, $\ker \phi^{\leq L}$ has positive dimension and contains a non-zero element h. Since $\ker \phi^{\leq L}$ is a subspace of $k[Y]^{\leq L}$, the degree of h can be at most L. □

For the sake of completeness, let us discuss the bit complexity of computing $D(n)$.

Lemma 15. *Let $d \in \mathbb{N}$ be constant. There is an algorithm that on input $n, m > 0$ with $m > n$ outputs $D \in \mathbb{N}$ s.t.*

$$\binom{m+D}{D} > \binom{n+dD}{dD}. \tag{1.90}$$

The bit complexity of this algorithm lies in $O(\log(m+D)^2 \cdot D^2)$.

Proof. We propose the following algorithm for computing D:

```
set $l := 1$
set $r := 1$
set $L := 0$

while $l \leq r$ do
    set $L := L + 1$
    set $l := l \cdot (m + L)$
    set $l := l/L$
    for $i = 1, \ldots, d$ do
        set $r := r \cdot (n + d(L-1) + i)$
        set $r := r/(d(L-1) + i)$
    end for
end while
return $L$
```

Informally, the algorithm computes for increasing $L = 1, \ldots, D$ the values

$$l := \binom{m+L}{L} \qquad \text{and} \qquad r := \binom{n+dL}{dL}. \tag{1.91}$$

It holds and outputs L if $l > r$. It is easy to see that the algorithm is correct. To estimates its bit security, we need to estimate the bit complexity in Lines 6, 7, 9 and 10.

In Lines 6 and 7, the algorithms first multiplies $l = \binom{m+L-1}{L-1}$ with $m+L$ and divides L from it. The bit complexity for both operations lies in

$$\begin{aligned} &O(\log(\binom{m+L-1}{L-1}) \cdot \log(m+L) + \log(\binom{m+L-1}{L-1} \cdot (m+L)) \cdot \log(L)) \\ \subseteq &O(\log(m+L)^2 \cdot L). \end{aligned}$$

Similarly, in Lines 9 and 10, the algorithm multiplies $r = \binom{n+d(L-1)+i-1}{d(L-1)+i-1}$ with $(n+d(L-1)+i)$ and divides $(d(L-1)+i)$ from it. The bit complexity for those operations lies in

$$\begin{aligned} &O(\log(\binom{n+d(L-1)+i-1}{d(L-1)+i-1}) \cdot \log(n+d(L-1)+i)) \\ &+ O(\log(\binom{n+d(L-1)+i-1}{d(L-1)+i-1} \cdot (n+d(L-1)+i)) \cdot \log(d(L-1)+i)) \\ \subseteq &O(\log(n+dL)^2 \cdot L). \end{aligned}$$

Since d is constant and $n < m$, $O(\log(n+dL)^2 \cdot L)$ is contained in $O(\log(m+L)^2 \cdot L)$.

Since the lines Lines 6, 7, 9 and 10 are repeated D and dD times, respectively, the total bit complexity is contained in

$$O(\log(m+D)^2 \cdot D^2). \qquad \square$$

We can now give an algorithm that computes an algebraic relationship h of degree D for a set of polynomials $f_1, \ldots, f_m$.

Algorithm 1 (Algebraic Relation Finder)**.** Let k be a field. The algorithm $\mathcal{B}$ gets as inputs a list of polynomials $f_1, \ldots, f_m \in k[X_1, \ldots, X_n]$ of degree d or a polynomial map $F : k^n \to k^m$ that consists of polynomials $f_1, \ldots, f_m$. We assume that $\mathcal{B}$ can automatically deduct the parameters $n, m, d \in \mathbb{N}$ from its input and that we have $m > n$.

$\mathcal{B}$ proceeds as follows:

Step 1: $\mathcal{B}$ computes the minimal $D \in \mathbb{N}$ s.t.

$$\binom{m+D}{D} > \binom{n+dD}{dD} \tag{1.92}$$

by using the algorithm from Lemma 15.

Step 2: $\mathcal{B}$ computes a matrix representation $M \in k^{\binom{n+dD}{dD} \times \binom{m+D}{D}}$ of the restricted map

$$\begin{aligned} \phi^{\leq D} : k[Y]^{\leq D} &\longrightarrow k[X]^{\leq dD} \qquad (1.93) \\ h(Y_1, \ldots, Y_m) &\longmapsto h(f_1, \ldots, f_m) \qquad (1.94) \end{aligned}$$

with respect to the monomial bases of $k[Y]^{\leq D}$ and $k[X]^{\leq dD}$.

Step 3: $\mathcal{B}$ uses Gaussian elimination to find a kernel vector of M and interprets it as an element h of $k[Y]^{\leq D}$ with respect to the monomial basis of $k[Y]^{\leq D}$. It outputs h.

From Lemma 14, we can deduce the correctness of $\mathcal{B}$:

Lemma 16. *$\mathcal{B}$ is correct, in the sense that for all $n, m, d \in \mathbb{N}$ with $m > n$, it will output an algebraic relationship h of degree D among its input $f_1, \ldots, f_m$, where we have*

$$D = \min\left\{ L \in \mathbb{N} \;\middle|\; \binom{n + d \cdot L}{d \cdot L} > \binom{m(n) + L}{L} \right\}. \tag{1.95}$$

Proof. Because of Lemma 15, $\mathcal{B}$ computes D correctly. Because of Lemma 14, the kernel of $\phi^{\leq D}$ is non-trivial.

Now, set $N_X := \binom{n+dD}{dD}$ and $N_Y := \binom{m+D}{D}$, and let $X^{\alpha(1)}, \ldots, X^{\alpha(N_X)}$ and $Y^{\beta(1)}, \ldots, Y^{\beta(N_Y)}$ be enumerations of the monomial bases of $k[X]^{\leq dD}$ and $k[Y]^{\leq D}$, respectively. Let M be the matrix representation of $\phi^{\leq D}$ with respect to the monomial bases. We have for each vector $w \in k^{N_Y}$

$$\phi^{\leq D}\left(\sum_{i=1}^{N_Y} w_i \cdot Y^{\beta(i)}\right) = \sum_{i=1}^{N_X} u_i \cdot X^{\alpha(i)}, \tag{1.96}$$

where $u = M \cdot w$. Let $v \in \ker M$ be the kernel vector found by $\mathcal{B}$ in Step 3. We have for v

$$\phi^{\leq D}\left(\sum_{i=1}^{N_Y} v_i \cdot Y^{\beta(i)}\right) = 0. \tag{1.97}$$

Hence, the polynomial

$$h := \sum_{i=1}^{N_Y} v_i \cdot Y^{\beta(i)} \tag{1.98}$$

that $\mathcal{B}$ outputs lies in the kernel of $\phi^{\leq D}$ and is an algebraic relation of degree $\leq D$. □

Lemma 17. *Let $m, n, d \in \mathbb{N}$ be s.t. d is constant and $m > n$.*

$\mathcal{B}$ makes $O(\log(m + D)^2 \cdot D^2)$ bit operations and

$$O\left(\binom{m+D}{D} \cdot \binom{n+dD}{dD}^2\right) \subseteq O\big(m^{3D}\big) \tag{1.99}$$

arithmetic operations over k.

Proof. The number of bit operations that $\mathcal{B}$ performs in Step 1 comes from Lemma 15.

In Step 2, $\mathcal{B}$ needs to compute a matrix M of shape $\binom{n+dD}{dD} \times \binom{m+D}{D}$ over k. Each column of M is the coefficient vector of the product of D polynomials of degree d over n variables. Computing this product costs $O\Big(\binom{n+dD}{dD}\Big)$ arithmetic operations over k. Since M has $\binom{m+D}{D}$ columns, $\mathcal{B}$ needs to make $\binom{m+D}{D} \cdot O\Big(\binom{n+dD}{dD}\Big)$ arithmetic operations to compute M.

Finally, since $\binom{m+D}{D} > \binom{n+dD}{dD}$, the Gaussian elimination of the matrix $M \in k^{\binom{N+dD}{dD}\times\binom{m+D}{D}}$ in Step 3 costs $O\Big(\binom{n+dD}{dD}^2 \cdot \binom{m+D}{D}\Big) \subseteq O\Big(\binom{m+D}{D}^3\Big)$ arithmetic operations.

In total, $\mathcal{B}$ makes $O\Big(\binom{m+D}{D} \cdot \binom{n+dD}{dD}^2\Big)$ arithmetic operations over k and $O\big(\log(m+D)^2 \cdot D^2\big)$ bit operations. □

We can now deduce that the algebraic relation h whose existence is claimed by Theorem 10 can be computed by $\mathcal{B}$ in time $O\big(m^{3D}\big)$.

Corollary 18. *In the situation of Theorem 10, Algorithm 1 computes on input $f_1, \ldots, f_m \in k[X_1, \ldots, X_n]^{\leq d}$ an algebraic relationship among $f_1, \ldots, f_m$ of degree $\leq D$ by making $O(\log(m+D)^2 \cdot D^2)$ bit operations and*

$$O\left(\binom{m+D}{D} \cdot \binom{n+dD}{dD}^2\right) \subseteq O\left(\binom{m+D}{D}^3\right) \subseteq O(m^{3D}) \tag{1.100}$$

arithmetic operations over k.

If $m \geq n^{1+e}$, then the number of arithmetic operations of Algorithm 1 over k lies in

$$O\left(n^{(1+e)\cdot 3 \cdot \left\lceil 2^{\frac{d}{d-1}} \cdot n^{1-\frac{e}{d-1}} \right\rceil}\right). \tag{1.101}$$

1.2.1 Reduced Algebraic Relations

Imagine we have a polynomial map $F : \mathbb{Z}_2^n \to \mathbb{Z}_2^m$ of constant degree d and use Algorithm 1 to compute an algebraic relation h for $\mathcal{F}$. Now, it may happen that h is of the shape $Y_i^2 - Y_i$ or, more generally, lies in the ideal $I_2 = (Y_1^2 - Y_1, \ldots, Y_m^2 - Y_m)$. This h will not be very useful for us, since it will vanish on any point $y \in \mathbb{Z}_2^m$. I.e., h will not be able to discern between points in the image of F and points outside the image of F.

We can fix this problem by adapting Algorithm 1 s.t. it automatically reduces domain and image values of ϕ modulo the ideals I_q where q is the size of k.

Let us first give the theoretical result of existence:

Theorem 19. *Let k be a field of finite size $q = q(n) > 1$. Further, let $d \in \mathbb{N}$ be constant and $m : \mathbb{N} \to \mathbb{N}$ be a function with*

$$m(n) \geq 2^{2d-1} \cdot d^{d-1} \cdot n. \tag{1.102}$$

Define the function $D : \mathbb{N} \to \mathbb{N}$ as follows

$$D(n) := \min\left\{L \in \mathbb{N} \;\middle|\; \dim_k k[Y]^{\leq L}/I_{q(n)} > \dim_k k[X]^{\leq dL}/I_{q(n)}\right\}. \tag{1.103}$$

Then, the following holds:

1. For $n \geq 2d$, we have

$$D(n) \leq \left\lceil 2^{\frac{d+1}{d-1}} \cdot \left(\frac{n^d}{m(n)}\right)^{1/(d-1)} \right\rceil. \tag{1.104}$$

2. *For all $n \in \mathbb{N}$ and $d > 1$, we have*

$$D(n) \geq \frac{1}{d} \cdot \left(\frac{n^d}{2^{d+1} \cdot m(n)} \right)^{1/(d-1)}. \tag{1.105}$$

3. *For each $n \in \mathbb{N}$ and each collection $f_1, \ldots, f_m \in k[X]^{\leq d}$ of $m(n)$ polynomials of degree d over n variables there exists a polynomial $h \in k[Y]$ with:*

$$h \notin I_q \subset k[Y], \tag{1.106}$$

$$\deg h \leq D(n), \tag{1.107}$$

$$h(f_1, \ldots, f_m) \in I_q \subset k[X]. \tag{1.108}$$

The proof of Theorem 19 is analogous to the proof of its unreduced version Theorem 10.

Lemma 20. *Let $d \in \mathbb{N}$ and let $m : \mathbb{N} \to \mathbb{N}$ s.t. $m(n) \geq 2^{2d-1} \cdot d^{d-1} \cdot n$ for all n. Then, we have for all integers $n \geq 2d$*

$$\dim_k k[Y]^{\leq L}/I_{q(n)} > \dim_k k[X]^{\leq dL}/I_{q(n)}, \tag{1.109}$$

where $L(n) = \left\lceil 2^{\frac{d+1}{d-1}} \left(\frac{n^d}{m} \right)^{\frac{1}{d-1}} \right\rceil$.

Proof. Because of Lemma 4, it suffices to prove

$$\binom{m(n)}{L(n)} > \binom{n + d \cdot L(n)}{d \cdot L(n)}. \tag{1.110}$$

We have already shown in the proof of Lemma 12 that the inequalities $n \geq 2d$ and $m \geq d^{d-1} \cdot 2^d \cdot 2^{d-1} \cdot n$ imply $2n \geq n + dL$.

We claim this time additionally that we have

$$m - L \geq \frac{m}{2}, \tag{1.111}$$

which is equivalent to $m \geq 2L$. Indeed, we have for each $d \in \mathbb{N}$

$$m \geq d^{d-1} \cdot 2^d \cdot 2^{d-1} \cdot n \geq 2n \geq 2d \cdot L. \tag{1.112}$$

Hence, $m \geq 2L$ and $m - L \geq m/2$ follow, respectively.

We can now repeat the strategy of Lemma 12. Note that the following inequalities are equivalent:

$$\binom{m}{L} > \binom{n + dL}{dL} \tag{1.113}$$

$$\iff \frac{m \cdots (m - L + 1)}{L!} > \frac{(n + dL) \cdots (n + 1)}{(dL)!} \tag{1.114}$$

$$\iff m \cdots (m - L + 1) \cdot (dL) \cdots (L + 1) > (n + dL) \cdots (n + 1) \tag{1.115}$$

Again, we lower bound the left-hand side and upper bound the right-hand side. For the left-hand side, we have

$$m \cdots (m - L + 1) > \left(\frac{m}{2} \right)^L, \tag{1.116}$$

$$(dL) \cdots (L + 1) > L^{(d-1)L}. \tag{1.117}$$

For the right-hand side, we have for n large enough

$$(n+dL)\cdots(n+1) \leq (n+dL)^{dL} \leq (2n)^{dL} = 2^{dL} \cdot n^{dL}. \tag{1.118}$$

Eq. (1.115) is now implied by the following inequalities

$$\left(\frac{m}{2}\right)^{L} \cdot L^{(d-1)L} \geq 2^{dL} \cdot n^{dL} \tag{1.119}$$

$$\iff \frac{m}{2} \cdot L^{d-1} \geq 2^{d} \cdot n^{d}. \tag{1.120}$$

Eq. (1.120) holds since we have $L \geq 2^{\frac{d+1}{d-1}} \cdot \left(\frac{n^d}{m}\right)^{\frac{1}{d-1}}$. □

Lemma 21. *Let $n, m, d, D \in \mathbb{N}$ with $d > 1$. Further, let k be a field of size q s.t.*

$$m \geq 2^{2d-1} \cdot d^{d-1} \cdot n, \tag{1.121}$$

$$\dim_k k[Y]^{\leq L}/I_q > \dim_k k[X]^{\leq dL}/I_q. \tag{1.122}$$

Then, we have

$$L \geq \frac{1}{d} \cdot \left(\frac{n^d}{2^{d+1} \cdot m}\right)^{1/(d-1)}. \tag{1.123}$$

Proof. First note that Eq. (1.122) and Lemma 4 imply together

$$\binom{m+L}{L} > \binom{n}{dL}. \tag{1.124}$$

Again, let us assume—for the sake of contradiction—that we have

$$L < \frac{1}{d}\left(\frac{n^d}{2^{d+1} \cdot m}\right)^{1/(d-1)}. \tag{1.125}$$

We claim that this implies

$$n - dL \geq \frac{n}{2}. \tag{1.126}$$

Indeed, we have

$$2dL < 2 \cdot \left(\frac{n^d}{2^{d+1} \cdot m}\right)^{1/(d-1)} \tag{1.127}$$

$$\leq \left(\frac{n^d}{2^2 \cdot m}\right)^{1/(d-1)} \tag{1.128}$$

$$\leq \left(\frac{n^d}{2^{2d+1} \cdot d^{d-1} \cdot n}\right)^{1/(d-1)} \tag{1.129}$$

$$\leq \left(\frac{1}{2^{2d+1} \cdot d^{d-1}}\right)^{1/(d-1)} \cdot n < n. \tag{1.130}$$

Further, we already showed in the proof of Lemma 13 that the assumption together with the requested inequalities imply

$$2m \geq m + L. \tag{1.131}$$

Now, we have the following equivalences:

$$\binom{m+L}{L} > \binom{n}{d \cdot L} \tag{1.132}$$

$$\iff (m+L)\cdots(m+1)\cdot(dL)\cdots(L+1) > n\cdots(n-dL+1). \tag{1.133}$$

The last inequality implies

$$(m+L)^L \cdot (dL)^{(d-1)L} > (n-dL)^{dL}. \tag{1.134}$$

By upper bounding the right-hand side and lower bounding the left-hand side, we get

$$(2m)^L \cdot (dL)^{(d-1)L} > \left(\frac{n}{2}\right)^{dL}. \tag{1.135}$$

We again take L-th roots and get

$$2m \cdot (dL)^{d-1} > \left(\frac{n}{2}\right)^{d}, \tag{1.136}$$

which is equivalent to

$$L > \frac{1}{d} \cdot \left(\frac{n^d}{2^{d+1}m}\right)^{1/(d-1)}. \tag{1.137}$$

Ergo, a contradiction! Hence, the claim of the lemma follows. □

To prove the existence of h, we first need to find a suitable dual morphism for the reduced polynomials $f_1, \ldots, f_m \in k[X]/I_q$ and prove its well-definedness. For this end, we show that the Frobenius map $x \mapsto x^q$ extends to a ring morphism on $k[X]$.

Lemma 22. *Let R be a ring of prime characteristic $p > 0$ and let $q = p^r$ for $r > 0$. For $x, y \in R$, we have*

$$(x+y)^q = x^q + y^q. \tag{1.138}$$

Proof. Since, q is a power of the characteristic p of R, it suffices to show that we have

$$(x+y)^p = x^p + y^p. \tag{1.139}$$

We can expand the power on the left-hand side as the sum

$$(x+y)^p = \sum_{i=0}^{p} \binom{p}{i} \cdot x^i \cdot y^{p-i}. \tag{1.140}$$

Since p is a prime, the coefficient $\binom{p}{i}$ is divisible by p for $i \in \{1, \ldots, p-1\}$ and must vanish in R. Hence, x^p and y^p are the only non-zero summands of the above sum. □

Lemma 23. *Let $f_1, \ldots, f_m \in k[X]$ be of degree d. Let ϕ be their corresponding dual morphism, i.e.,*

$$\begin{aligned} \phi : k[Y] &\longrightarrow k[X] && (1.141) \\ Y_i &\longmapsto f_i. && (1.142) \end{aligned}$$

We have

$$\phi((Y_1^q - Y_1, \ldots, Y_m^q - Y_m)) \subseteq (X_1^q - X_1, \ldots, X_m^q - X_m). \tag{1.143}$$

Proof. It suffices to show that $\phi(Y_1^q - Y_1) = f_1^q - f_1$ lies in $I_q \subset k[X]$. For this end, we will show f_1^q equals f_1 modulo the field equations of k. Because of Lemma 22, the Frobenius-like map

$$\begin{aligned} \rho : k[X] &\longrightarrow k[X] && (1.144) \\ g &\longmapsto g^q && (1.145) \end{aligned}$$

is a ring morphism on $k[X]$. Write f_1 as

$$f_1 = \sum_{\alpha} c_\alpha \cdot X^\alpha \tag{1.146}$$

for $c_\alpha \in k$. Since ρ is a ring morphism, we have

$$f_1^q = \sum_{\alpha} c_\alpha^q \cdot X^{q\alpha} = \sum_{\alpha} c_\alpha \cdot X^{q\alpha}, \tag{1.147}$$

where we have $c_\alpha = c_\alpha^q$, since k is of size q. Now, the monomial $X^{q\alpha}$ can be reduced modulo I_q to X^α. This implies

$$f_1^q \equiv \sum_{\alpha} c_\alpha \cdot X^\alpha = f_1 \mod I_q. \tag{1.148}$$

Hence, $f_1^q - f_1 \in I_q$. It follows that we have $\phi(I_q) \subseteq I_q$. □

Definition 12. Given polynomials $f_1, \ldots, f_m \in k[X]/I_q$, we define their **dual morphism** by

$$\begin{aligned} \phi : k[Y_1, \ldots, Y_m]/I_q &\longrightarrow k[X_1, \ldots, X_n]/I_q && (1.149) \\ Y_j &\longmapsto f_j(X) + I_q. && (1.150) \end{aligned}$$

Because of Lemma 23, ϕ is well-defined.

Lemma 24. *Let $f_1, \ldots, f_m \in k[X]/I_q$ be of degree $\leq d$, and let $\phi : k[Y]/I_q \to k[X]/I_q$ be their dual morphism.*

If we have

$$\binom{m}{L} > \binom{n + dL}{dL} \tag{1.151}$$

for some $L \in \mathbb{N}$, then $\ker \phi$ must contain a non-zero element of degree L.

Proof. The proof of this lemma follows analogously the proof of Lemma 12. We note that we can restrict ϕ on the space of polynomials of degree $\leq L$ modulo I_q and get a linear map.

$$\phi^{\leq L} : k[Y]^{\leq L}/I_q \longrightarrow k[X]^{\leq dL}/I_q. \tag{1.152}$$

Since we have for the dimensions of domain and codomain

$$\dim_k k[Y]^{\leq L}/I_q \geq \binom{m}{L} \qquad \text{and} \qquad \dim_k k[X]^{\leq dL}/I_q \leq \binom{n+dL}{dL}, \tag{1.153}$$

it follows $\ker \phi^{\leq L} \neq 0$. □

Proof Theorem 19. It remains to prove the third claim of Theorem 19. For this end, let $f_1, \dots, f_m \in k[X]$ be of degree d. Denote their residue classes modulo I_q by $f'_1, \dots, f'_m \in k[X]/I_q$, and let $\phi' : k[Y]/I_q \to k[Y]/I_q$ be the dual morphism of $f'_1, \dots, f'_m$. Because of Lemma 24, $\ker \phi'$ has a non-trivial element h' of degree $\leq L$. Now, there must exist an element $h \in k[Y]$ of degree $\leq L$ s.t. $h + I_q = h'$. Since $h' \neq 0$, h does not lie in I_q. We need to prove that

$$h(f_1, \dots, f_m) \in I_q. \tag{1.154}$$

Denote by ϕ the dual morphism of $f_1, \dots, f_m$ and note that the following diagram commutes

$$\begin{tikzcd}
k[Y] \arrow[r, "\phi"] \arrow[d, "\pi_Y"'] & k[X] \arrow[d, "\pi_X"] \\
k[Y]/I_q \arrow[r, "\phi'"'] & k[X]/I_q
\end{tikzcd}$$

where π_X, π_Y are the canonical projections. Hence, we have

$$h(f_1, \dots, f_m) + I_q = \pi_X(\phi(h)) = \phi'(\pi_Y(h)) = \phi'(h') = 0.$$ □

We will now give the algorithm for finding the reduced algebraic relationship h of Theorem 19:

Algorithm 2 (Reduced Algebraic Relation Finder)**.** Let k be a finite field of size q. The algorithm $\mathcal{B}$ gets as inputs a list of polynomials $f_1, \dots, f_m \in k[X_1, \dots, X_n]$ of degree d or a polynomial map $F : k^n \to k^m$ that consists such polynomials. $\mathcal{B}$ expects that we have $m > n$.

$\mathcal{B}$ proceeds as follows:

Step 1: $\mathcal{B}$ computes the minimal $D \in \mathbb{N}$ s.t.

$$\binom{m}{D} > \binom{n+dD}{dD}. \tag{1.155}$$

Step 2: $\mathcal{B}$ computes a matrix representation M of the restricted map

$$\phi^{\leq D} : k[Y]^{\leq D}/I_q \longrightarrow k[X]^{\leq dD}/I_q \tag{1.156}$$
$$h(Y_1, \dots, Y_m) \longmapsto h(f_1, \dots, f_m) + I_q. \tag{1.157}$$

Let us explain this step in more detail: $\mathcal{B}$ computes monomial bases of $k[Y]^{\leq D}/I_q$ and $k[X]^{\leq dD}/I_q$, which are given by

$$\{X^\alpha \mid \alpha \in \{0, \ldots, q-1\}^n, ||\alpha||_1 \leq dD\}, \tag{1.158}$$

$$\{Y^\beta \mid \beta \in \{0, \ldots, q-1\}^m, ||\beta||_1 \leq D\}. \tag{1.159}$$

Now, for each column of M, $\mathcal{B}$ computes a product $f_{j_1} \cdots f_{j_{D'}}$, $D' \leq D$, of degree-d polynomials modulo I_q. $\mathcal{B}$ does this by computing this product over $k[Y]$ and reducing each variable-power $X_i^{a+b\cdot(q-1)}$ to X_i^a, whenever $a \in \{0, \ldots, q-1\}$ and $b > 0$.

Step 3: $\mathcal{B}$ uses Gaussian elimination to find a kernel vector of M and interprets it as an element h of $k[Y]^{\leq D}$ with respect to the monomial basis of Eq. (1.159). It outputs h.

The correctness of $\mathcal{B}$ follows from Lemma 24, and we have for the time complexity of $\mathcal{B}$ the same estimates as for the time complexity of its non-reduced version Algorithm 1. Hence, we state without proof:

Proposition 25. *Algorithm 2 is correct, in the sense that, on input $f_1, \ldots, f_m \in k[X]^{\leq d}$ with $m > 2^{2d-1} \cdot d^{d-1} \cdot n$ and $n \geq 2d$, it will always output a polynomial $h \in k[Y]$ s.t.*

$$h \notin I_q \subset k[Y], \tag{1.160}$$

$$\deg h \leq \left\lceil 2^{\frac{d+1}{d-1}} \cdot \left(\frac{n^d}{m}\right)^{1/(d-1)} \right\rceil, \tag{1.161}$$

$$h(f_1, \ldots, f_m) \in I_q \subset k[X]. \tag{1.162}$$

Further, it performs $O(\log(m+D)^2 \cdot D^2)$ bit operations and $O\Big(\binom{m+D}{D}^3\Big) \subset O(m^{3D})$ arithmetic operations over k.

Corollary 26. *If $m \geq n^{1+e}$, then Algorithm 2 performs*

$$O\left(n^{(1+e)\cdot 3\cdot \left\lceil 2^{\frac{d+1}{d-1}} \cdot n^{1-e/(d-1)} \right\rceil}\right) \tag{1.163}$$

arithmetic operations over k and outputs an element of degree $O(n^{1-e/(d-1)})$.

Remark 1. Theorem 19 and Algorithm 2 handle the case of algebraic PRGs $\mathcal{F} : k^n \to k^m$ over small fields (i.e., $\#k \in o(n)$). However, one might also consider the case of an algebraic PRG

$$\mathcal{F} : \{0,1\}^n \to \mathbb{Z}_q^m \tag{1.164}$$

of constant degree d. I.e., the seed of $\mathcal{F}$ is a uniformly random bit string, however, the output of $\mathcal{F}$ is supposed to imitate a uniformly random vector of $\mathbb{Z}_q$ (for $q \geq n$). In this case, one can adapt the Algorithms 1 and 2 by computing the matrix representation of the dual map

$$\phi^{\leq D} : k[Y] \to k[X]/I_2 \tag{1.165}$$

for the value

$$D = \min \left\{ L \in \mathbb{N} \;\middle|\; \dim_k k[Y]^{\leq L} = \binom{m+L}{L} > \binom{n}{dL} = \dim_k k[X]^{\leq dL}/I_2 \right\}.$$

While D has the same asymptotic behaviour as the degree bounds given in Theorems 10 and 19, it will be by some constant smaller than the degree bounds computed by the Algorithms 1 and 2.

In general, if one can restrict the domain and codomain of a PRG $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$ over $k = (k_n)_n$, one can accelerate and improve the proposed algorithms. Let us assume that there are families of ideals $I = (I_n)_n$ and $J = (J_n)_n$

$$I_n \subseteq k_n[X_1, \ldots, X_n], \tag{1.166}$$

$$J_n \subseteq k_n[Y_1, \ldots, Y_{m(n)}]. \tag{1.167}$$

s.t. the following conditions are fulfilled for each $n \in \mathbb{N}$:

1. Each element $x \leftarrow \mathcal{X}_n$ lies in the variety $V(I_n)$ and each element $y \leftarrow \mathcal{Y}_n$ lies in the variety $V(J_n)$.
2. Groebner bases for the ideals I_n, J_n are known with respect to degree-preserving monomial orderings.
3. For each $F \leftarrow \mathcal{F}_n$, we have for its dual morphisms $\phi_F : k_n[Y] \to k_n[X]$

$$\phi_F(J_n) \subseteq I_n. \tag{1.168}$$

Then, one can adapt Algorithm 2 s.t. on input $F \leftarrow \mathcal{F}_n$ it computes a matrix representation of

$$\widetilde{\phi_F}^{\leq L} : k_n[Y]^{\leq L}/J_n \longrightarrow k_n[X]^{\leq dL}/I_n \tag{1.169}$$

and outputs an element $h \in k_n[Y]$ with

$$h \notin J_n, \tag{1.170}$$

$$h(F) \in I_n, \tag{1.171}$$

$$\deg h \leq \min \left\{ L \in \mathbb{N} \;\middle|\; \dim_k k_n[Y]^{\leq L}/J_n > \dim_k k_n[X]^{\leq dL}/I_n \right\}. \tag{1.172}$$

1.2.2 On the Optimality of $D(n)$

For some constant $d \in \mathbb{N}$ and a function $m : \mathbb{N} \to \mathbb{N}$ with $m(n) > n$, let $D : \mathbb{N} \to \mathbb{N}$ be the function from Theorem 10, i.e.,

$$D(n) := \min \left\{ L \in \mathbb{N} \;\middle|\; \binom{m(n)+L}{L} > \binom{n+dL}{dL} \right\}. \tag{1.173}$$

Note that Algorithm 1 on input $f_1, \ldots, f_m \in k[X]$ always computes a matrix representation of the dual morphism $\phi^{\leq D(n)}$ up to degree $D(n)$. However, there are a lot of examples of polynomials $f_1, \ldots, f_m \in k[X]$ of degree d s.t. the corresponding dual morphism ϕ has non-trivial kernel elements of a degree that is substantially smaller than $D(n)$. Hence, a potential optimization could be to modify Algorithm 1 (and Algorithm 2) s.t. for increasing $L = 1, \ldots, D(n)$, it tries to compute a non-trivial kernel element of ϕ of degree L. If an algebraic relationship of degree $L \ll D(n)$ exists, then this algorithm terminates faster than Algorithm 1. Let us give a formal description of this algorithm:

Algorithm 3 (Optimized Relation Finder)**.** The algorithm $\mathcal{B}_{\mathsf{opt}}$ gets as inputs a list of polynomials $f_1, \ldots, f_m \in k[X_1, \ldots, X_n]$ of degree d.

$\mathcal{B}_{\mathsf{opt}}$ proceeds as follows:

```
Step 1: For each $L = 1, 2, \ldots$, $\mathcal{B}_{\mathsf{opt}}$ does the following:
    Step 1: $\mathcal{B}_{\mathsf{opt}}$ computes a matrix representation $M_L$ of
            $\phi^{\leq L} : k[Y]^{\leq L} \longrightarrow k[X]^{\leq dL}$    (1.174)
            $Y_i \longmapsto f_i.$    (1.175)
    Step 2: If $M_L$ has a non-zero kernel vector, $\mathcal{B}_{\mathsf{opt}}$ finds it by Gaussian elimination
            and outputs the corresponding polynomial $h \in \ker \phi^{\leq L}$.
```

It is easy to see that $\mathcal{B}_{\mathsf{opt}}$ is correct when $m > n$ and that its runtime lies in $O(D \cdot m^{3D})$.

In this subsection, we want to investigate if $\mathcal{B}_{\mathsf{opt}}$ is faster than $\mathcal{B}$ in the average case. As we already stated in the beginning, it is easy to find examples of functions $f_1, \ldots, f_m$ for which $\mathcal{B}_{\mathsf{opt}}$ is significantly faster than $\mathcal{B}$. An (almost) counter-example is given by Zichron [Zic17] in his master thesis:

Lemma 27 (Theorem 1.8 in [Zic17])**.** *Let k be a field, $d \geq 2$ and $m = \lceil n^{1+e} \rceil$ for $e \in (0, 0.5)$ constant.*

There is a degree-d map $F : k^n \to k^m$ s.t. the minimal degree of an algebraic relation of F lies in $\Omega(n^{1-e/(d-2.1)})$, i.e.

$$\min \{\deg h \mid h \in k[Y], h \neq 0, h \circ F = 0\} \in \Omega(n^{1-e/(d-2.1)}). \tag{1.176}$$

Note that the asymptotic class $\Omega(n^{1-e/(d-2.1)})$ is close to the asymptotic class $\Theta(n^{1-e/(d-1)})$ of $D(n)$. However, the counter-example of Zichron is a special local PRG. What can we say about the performance of $\mathcal{B}_{\mathsf{opt}}$ in the *average* case, i.e., when the coefficients of the polynomials $f_1, \ldots, f_m \leftarrow k[X]^{\leq d}$ are sampled uniformly and independently at random over an exponentially large field k? I suspect that in this case $D(n)$ is almost always optimal, i.e., $\phi^{\leq L}$ is injective whenever $L < D(n)$. To make this formal, we can follow the approach of Diem [Die04]. To prove lower bounds for the XL algorithm [CKPS00], Diem investigated the behaviour of systems of *generic* polynomials. Let us introduce them formally:

Definition 13. Let k be a field and $n, m, d \in \mathbb{N}$. Let $\alpha(1), \ldots, \alpha(\binom{n+d}{d}) \in \mathbb{N}_0^n$ be an enumeration of all multi-indices in the set

$$\{\beta \in \mathbb{N}_0^n \mid d \geq ||\beta||_1\}. \tag{1.177}$$

For each $j \in [m]$ and $i \in [\binom{n+d}{d}]$, let $C_{j,i}$ be a fresh new variable, and set

$$K := k(C) = k((C_{j,i})_{j \in [m], i \in [\binom{n+d}{d}]}) \tag{1.178}$$

to be the field that is generated by k and the variables $(C_{j,i})_{j \in [m], i \in [\binom{n+d}{d}]}$.

For $j \in [m]$, we define the j-th **generic polynomial** by

$$g_j := \sum_{i=1}^{\binom{n+d}{d}} C_{j,i} \cdot X^{\alpha(i)} \in K[X_1, \ldots, X_n], \tag{1.179}$$

and we define the **generic map** of degree d by

$$\begin{aligned} G : K^n &\longrightarrow K^m & (1.180)\\ x &\longmapsto (g_1(x), \dots, g_m(x)). & (1.181) \end{aligned}$$

Further, denote by ϕ_{gen} the *generic dual morphism* that is given by

$$\begin{aligned} \phi_{\mathsf{gen}} : K[Y_1, \dots, Y_m] &\longrightarrow K[X_1, \dots, X_n] & (1.182)\\ Y_i &\longmapsto g_i(X). & (1.183) \end{aligned}$$

The generic polynomial map G idealizes the behaviour of a uniformly random map $F : k^n \to k^m$ of degree d. Indeed, we have the following:

Lemma 28. *Draw $f_1, \dots, f_m \leftarrow k[X]^{\leq d}$ uniformly and independently at random and consider their dual morphism*

$$\begin{aligned} \phi : k[Y] &\longrightarrow k[X] & (1.184)\\ Y_i &\longmapsto f_i. & (1.185) \end{aligned}$$

For each $L \in \mathbb{N}$, if $\phi_{\mathsf{gen}}^{\leq L}$ is injective, then we have

$$\Pr_{f_1,\dots,f_m \leftarrow k[X]^{\leq d}} \left[\phi^{\leq L} \textit{ is injective}\right] \geq 1 - \frac{L \cdot \binom{m+L}{L}}{\#k}. \qquad (1.186)$$

Proof. Assume that the restriction of the dual morphism $\phi_{\mathsf{gen}}^{\leq L}$ of the generic polynomials $g_1, \dots, g_m$ is injective, and let $M \in K^{\binom{n+dL}{dL} \times \binom{m+L}{L}}$ be a matrix representation of $\phi_{\mathsf{gen}}^{\leq L} : K[Y]^{\leq L} \to K[X]^{\leq dL}$. Note that each column of M contains the coefficient vector of a product $g_{i_1} \cdots g_{i_{L'}} \in k[C, X]$, $L' \leq L$. Hence, the entries of M are polynomials over k in the variables $(C_{j,i})_{j,i}$ of degree $\leq L$.

Since $\phi_{\mathsf{gen}}^{\leq L}$ is injective, M must have full rank $\binom{m+L}{L}$. Hence, M has a regular submatrix $S \in K^{\binom{m+L}{L} \times \binom{m+L}{L}}$. The determinant of S is a non-zero element of $k[C]$. Let us denote this element by $h(C) := (\det S)(C)$. For the degree of h over $(C_{j,i})_{j,i}$, we have

$$\deg h \leq \binom{m+L}{L} \cdot L, \qquad (1.187)$$

since the determinant of a $\binom{m+L}{L} \times \binom{m+L}{L}$-matrix is a polynomial of degree $\binom{m+L}{L}$ over the entries of the matrix, and since the entries of S are polynomials of degree $\leq L$ over the variables $(C_{j,i})_{j,i}$.

Now, draw elements $c_{j,i} \leftarrow k$ for $j \in [m], i \in [\binom{n+d}{d}]$ uniformly and independently at random, and set

$$f_j := \sum_{i=1}^{\binom{n+d}{d}} c_{j,i} \cdot X^{\alpha(i)}. \qquad (1.188)$$

Then, the polynomials $f_1, \dots, f_m$ are distributed uniformly and independently over $k[X]^{\leq d}$.

Further, introduce the following ring morphism

$$E : k[C] \longrightarrow k \tag{1.189}$$

$$C_{j,i} \longmapsto c_{j,i}. \tag{1.190}$$

E replaces the generic coefficients $C_{j,i}$ by the random coefficients $c_{j,i}$. Since E is a ring morphism, we can extend it to maps $E : k[C, X] \to k[X]$ and $E : (k[C])^{\binom{n+dL}{dL} \times \binom{m+L}{L}} \to k^{\binom{m+L}{L} \times \binom{m+L}{L}}$ by applying E coefficient- and entry-wise. Note that we have

$$E(g_j) = f_j. \tag{1.191}$$

Further, $E(M)$ is the matrix representation of the dual morphism $\phi : k[Y] \to k[X]$ that belongs to the polynomials $f_1, \ldots, f_m$. We claim that $E(M)$ is with high probability regular. Indeed, since S is a square-submatrix of M, $E(S)$ is a square-submatrix of $E(M)$. For the determinant of $E(S)$, we have

$$\det(E(S)) = E(\det(S)) = E(h) = h(c). \tag{1.192}$$

Since h is a non-zero polynomial of degree $L \cdot \binom{m+L}{L}$, the Schwartz-Zippel Lemma 2 implies

$$\Pr_{c \leftarrow k^{m \times \binom{n+d}{d}}} [h(c) \neq 0] \geq 1 - \frac{L \cdot \binom{m+L}{L}}{\#k}. \tag{1.193}$$

Hence, with probability $\geq 1 - \frac{L \cdot \binom{m+L}{L}}{\#k}$ the matrix $E(M)$ must have rank $\binom{m+L}{L}$. Since $E(M)$ having full rank implies $\phi^{\leq L}$ being injective, the claim follows. □

Lemma 28 implies that for large fields ($\#k \geq 2^n$, e.g.) random polynomials $f_1, \ldots, f_m \leftarrow k[X]^{\leq d}$ will not have an algebraic relationship of degree L if $\phi_{\mathsf{gen}}^{\leq L}$ is injective. If we could show that $\phi_{\mathsf{gen}}^{\leq L}$ is injective whenever $L < D(n)$, then this would imply that $D(n)$ is optimal for random systems of multivariate polynomials and would give us a lower bound for the algorithms and attacks presented in this chapter. Unfortunately, the injectivity of ϕ_{gen} is not clear. We ask here the following question:

Question 3. *Let k be a field and let $n, m, d \in \mathbb{N}$ with $m > n$. What is the maximum $L \in \mathbb{N}$ s.t.*

$$\phi_{\mathsf{gen}}^{\leq L} : k[Y]^{\leq L} \to k[X]^{\leq dL} \tag{1.194}$$

is injective?

Note that lower bounds for L in Question 3 can be given by finding examples of polynomials $f_1, \ldots, f_m \in k[X]$ s.t. the corresponding dual morphism $\phi^{\leq L}$ is injective. Indeed, whenever $\phi^{\leq L}$ is injective for a specific set of non-generic maps $f_1, \ldots, f_m$, the generic morphism $\phi_{\mathsf{gen}}^{\leq L}$ must be injective, too. Hence, Zichron's Lemma 27 implies that for $m \geq n^{1+e}$ the maximum L must lie in $\Omega(n^{1-e/(d-2.1)})$.

1.3 Algebraic Attacks on PRGs

Based on the algebraic relation finders of Section 1.2, we will give here adversaries for the pseudorandomness of algebraic PRGs $\mathcal{F} : k^n \to k^m$. In essence, each attack here uses Algorithm 1 or Algorithm 2 to compute an algebraic relation h of $F \leftarrow \mathcal{F}$, and then applies h on $y \in k^m$. If $y = F(x)$ for some $x \in k^n$, we must have $h(y) = h(F(x)) = 0$, since h is an algebraic relation of the polynomials computing each output value of F. Hence, the adversary claims that y is not truly random if $h(y) = 0$. Otherwise, if $y \leftarrow k^m$ is distributed uniformly at random, then we will give bounds for the probability of h vanishing at y. This attack idea yields a high advantage when k is of sufficient size, i.e., $\#k \geq n$, and a subexponential time complexity when $\mathcal{F}$ is of poly-stretch $m \geq n^{1+e}$, $e > 0$ constant.

In cases of small fields ($\#k \in o(n)$), we will adapt the attack in Section 1.3.2 s.t. h is reduced modulo the field equations of k. This yields an attack of the same time complexity, whose advantage is non-trivial, but subexponentially small. To get an adversary of high advantage against PRGs over small fields, we will in Section 1.3.3 extend the field k to a larger field $\overline{k} \supset k$ and change the PRG $\mathcal{F} : k^n \to k^m$ to a PRG $\mathcal{F}' : k^n \to \overline{k}^{m'}$ for $m' \approx m/\log(n)$. This will allow us to apply the basic algebraic attack for large fields again. Since we reduce the stretch of the PRG by a logarithmic factor, the time complexity of this attack will grow by a logarithmic factor in the exponent.

Remark 2. We will not make any assumptions about the distributions $\mathcal{F} = (\mathcal{F}_n)_n$ here. In fact, each attack here works even in the *worst-case*, i.e., the advantage guarantees of Theorems 29, 31 and 34 do hold for every fixed polynomial map $F : k^n \to k^m$ of degree d.

For the ease of notation, we state all our results here for PRGs $\mathcal{F} : k^n \to k^m$ and $\mathcal{F} : S^n \to T^m$ where seeds are distributed uniformly in k^n and S^n, for subsets $S, T \subset k$, respectively. However, we will at no point make use of the distribution of seeds. In fact, the results of Theorems 29, 31 and 34 do hold for PRGs $\mathcal{F} : \mathcal{X} \to k^m$ and $\mathcal{F} : \mathcal{X} \to T^m$, respectively, where $\mathcal{X}$ may be any distribution over k^n.

1.3.1 Basic Attack

Let k be a field, $S, T \subset k$ subsets and consider an algebraic PRG

$$\mathcal{F} : S^n \to T^m \tag{1.195}$$

of constant degree $d \in \mathbb{N}$. Proceeding from Algorithm 1 for computing algebraic relations, we will give here a simple first algorithm for attacking the pseudorandomness of $\mathcal{F}$.

Algorithm 4 (Basic Algebraic Attack)**.** Let $\mathcal{B}$ be the algebraic relation finder from Algorithm 1. We will give here an adversary $\mathcal{A} = (\mathcal{B}, \mathcal{A}_{\mathsf{on}})$ for the pseudorandomness Game 2 of a PRG $\mathcal{F} : S^n \to T^m$ of constant degree $d \in \mathbb{N}$ and stretch $m > n$ over a field k.

The offline part of $\mathcal{A}$ is given by $\mathcal{B}$. The online part of $\mathcal{A}$ is given by the following algorithm $\mathcal{A}_{\mathsf{on}}$:

Step 1: $\mathcal{A}_{\mathsf{on}}$ receives the tuple (F, y, h) from the challenger $\mathcal{C}$ of Game 2, where $F \leftarrow \mathcal{F}_n$ has been sampled by $\mathcal{C}$ and $h \in k[Y]$ is a hint generated by $\mathcal{B}$ on input h. Note that $y \in T_n^m$ is either an image $F(y) = x$ for $x \leftarrow S_n^n$, if $b = 0$, or a random point $y \leftarrow T_n^m$, if $b = 1$.

Step 2: $\mathcal{A}_{\mathsf{on}}$ evaluates h on y, i.e., it computes

$$z := h(y) \in k. \tag{1.196}$$

If $z = 0$, then $\mathcal{A}_{\mathsf{on}}$ outputs 0. Otherwise, it outputs 1.

While Algorithm 4 is quite simple, it gives a refutation algorithm for $\mathcal{F}$ with high advantage and subexponential runtime if the set T of random target values is large enough and $\mathcal{F}$ of poly-stretch.

Theorem 29. *Let k be a field with subsets $S, T \subseteq k$, and let $\mathcal{F}$ be a PRG*

$$\mathcal{F} : S^n \to T^m \tag{1.197}$$

of constant degree $d \in \mathbb{N}$ and stretch $m \geq 2^{2d-1} \cdot d^{d-1} \cdot n$.

Then, adversary $\mathcal{A}$ from Algorithm 4 has an advantage of

$$\mathsf{adv}_{\mathcal{F}}^{\mathsf{PRG}}(\mathcal{A}) \geq 1 - \left\lceil \left(\frac{(2n)^d}{m} \right)^{\frac{1}{d-1}} \right\rceil \cdot \frac{1}{\#T_n} \tag{1.198}$$

in Game 2, a total time complexity of $O(\log(m)^2 \cdot n^2)$ bit operations and

$$O\left(m^{3 \cdot \left\lceil 2^{\frac{d}{d-1}} \cdot \left(n^d/m\right)^{1/(d-1)} \right\rceil} \right) \tag{1.199}$$

arithmetic operations over k, and an online time complexity of

$$O\left(n \cdot m^{\left\lceil 2^{\frac{d}{d-1}} \cdot \left(n^d/m\right)^{1/(d-1)} \right\rceil} \right). \tag{1.200}$$

Corollary 30. *If $m \geq n^{1+e}$, for a constant $e > 0$, and $\#T_n \in \omega(n^{1-e/(d-1)})$, then adversary $\mathcal{A}$ from Algorithm 4 has a high advantage $\geq 1 - o(1)$ in the pseudorandomness Game 2 of $\mathcal{F} : S^n \to T^m$, a total time complexity of*

$$O\left(n^{3 \cdot (1+e) \cdot \left\lceil 2^{\frac{d}{d-1}} \cdot n^{1-e/(d-1)} \right\rceil} \right) \subset n^{O(n^{1-e/(d-1)})} \tag{1.201}$$

and an online time complexity of

$$O\left(n^{1+(1+e) \cdot \left\lceil 2^{\frac{d}{d-1}} \cdot n^{1-e/(d-1)} \right\rceil} \right) \subset n^{O(n^{1-e/(d-1)})}. \tag{1.202}$$

Proof Theorem 29. Let $n, d \in \mathbb{N}$ and $m \geq 2^{2d-1} \cdot d^{d-1} \cdot n$. Sample a polynomial map $F \leftarrow \mathcal{F}$ of type $k^n \to k^m$ and degree d and let $f_1, \ldots, f_m \in k[X]$ be the polynomials that compute each output value of F.

In Game 2, $\mathcal{B}$ receives F from the challenger and computes, according to Corollary 18, an algebraic relation $h \in k[Y]$ of $f_1, \dots, f_m$ of degree at most

$$D \le \left\lceil \left(\frac{(2n)^d}{m} \right)^{\frac{1}{d-1}} \right\rceil . \tag{1.203}$$

$\mathcal{B}$ sends h to the challenger who sends the tuple (F, y, h) to $\mathcal{A}_{\mathsf{on}}$ which outputs zero iff $h(y) = 0$

Let $b \leftarrow \{0, 1\}$ be the bit drawn by the challenger. If $b = 0$, then we have $y = F(x)$ for $x \leftarrow S_n^n$. Since h is an algebraic relation among $f_1, \dots, f_m$, we have

$$h(y) = h(F(x)) = h(f_1(x), \dots, f_m(x)) = (h(f_1(X), \dots, f_m(X)))(x) = (0)(x) = 0.$$

Hence, $\mathcal{A}_{\mathsf{on}}$ will always output 0 on input $(F, F(x), h)$. I.e.,

$$\Pr_{\substack{F \leftarrow \mathcal{F}_n \\ h \leftarrow \mathcal{B}(F) \\ x \leftarrow S^n}} [\mathcal{A}_{\mathsf{on}}(F, F(x), h) = 0] \Pr_{\substack{F \leftarrow \mathcal{F}_n \\ h \leftarrow \mathcal{B}(F) \\ x \leftarrow S^n}} [h(F(x)) = 0] = 1. \tag{1.204}$$

If $b = 1$, then $y \leftarrow T^m$ is chosen uniformly at random, while $h \in k[Y]$ is some polynomial of degree D that is independent of y. According to the Schwartz-Zippel Lemma 2, we have

$$\Pr[h(y) = 0] \le \frac{\deg h}{\#T_n} \le \frac{D}{\#T_n}. \tag{1.205}$$

It follows

$$\Pr_{\substack{F \leftarrow \mathcal{F}_\lambda \\ h \leftarrow \mathcal{B}(F) \\ y \leftarrow \mathcal{Y}_\lambda}} [\mathcal{A}_{\mathsf{on}}(F, y, h) = 1] = \Pr_{\substack{F \leftarrow \mathcal{F}_\lambda \\ h \leftarrow \mathcal{B}(F) \\ y \leftarrow \mathcal{Y}_\lambda}} [h(y) \ne 0] \ge 1 - \frac{D}{\#T_n}. \tag{1.206}$$

Now, for the advantage of $\mathcal{A}$ we have

$$\begin{aligned} \mathsf{adv}_{\mathcal{F}}^{\mathsf{PRG}}(\mathcal{A}) = & \Pr_{\substack{F \leftarrow \mathcal{F}_\lambda \\ h \leftarrow \mathcal{B}(F) \\ x \leftarrow \mathcal{X}_\lambda}} [\mathcal{A}_{\mathsf{on}}(F, F(x), h) = 0] + \Pr_{\substack{F \leftarrow \mathcal{F}_\lambda \\ h \leftarrow \mathcal{B}(F) \\ y \leftarrow \mathcal{Y}_\lambda}} [\mathcal{A}_{\mathsf{on}}(F, y, h) = 1] - 1 \\ \ge & 1 + 1 - \frac{D}{\#T_n} - 1 = 1 - \frac{D}{\#T_n}. \end{aligned}$$

The offline complexity of $\mathcal{A}$ equals the time complexity of $\mathcal{B}$, which is given by Corollary 18.

The online complexity of $\mathcal{A}$ is given by the time complexity of $\mathcal{A}_{\mathsf{on}}$, which is dominated by evaluating a polynomial of degree $\le D$ at m elements of k. This task can be performed by $D \cdot \binom{m+D}{D} \le D \cdot m^D$ arithmetic operations over k. □

1.3.2 Reduced Algebraic Attack

The advantage of Algorithm 4 is lower bounded by

$$\ge 1 - O\left(\frac{(n^d/m)^{1/(d-1)}}{\#T_n} \right). \tag{1.207}$$

When we have a superlinear stretch of $m \in \omega(n)$, then the numerator $(n^d/m)^{1/(d-1)}$ lies in $o(n)$, and it suffices that T grows at least linearly to guarantee a high advantage. However, in the case of a binary PRG

$$\mathcal{F} : \{0,1\}^n \to \{0,1\}^m \tag{1.208}$$

of low degree over $\mathbb{Z}_2$, the size of $T_n = \{0,1\}$ is always 2, and we do not get any guarantee for Algorithm 4. A first fix for this problem is to replace the offline algorithm by its reduced version Algorithm 2. In the case of $k = \mathbb{Z}_2$, this would lead to an attack algorithm that always uses an algebraic relation $h \in \mathbb{Z}_2[Y]$ that is non-zero modulo $I_2 = (Y_1^2 - Y_1, \ldots, Y_m^2 - Y_m)$. Lemma 5 implies for such an h a probability of $2^{-\deg h}$ to not vanish on a uniformly random bit vector. This leads to a distinguishing advantage of $2^{-\deg h}$, which is larger than the trivial distinguishing advantage of 2^{-n}, however subexponentially small. The next theorem makes these observations more precise:

Theorem 31. *Let k be a* finite *field k, and let $\mathcal{F} : k^n \to k^m$ be a PRG of constant degree d over k.*

Consider the adversary $\mathcal{A} = (\mathcal{B}, \mathcal{A}_{\mathsf{on}})$ for the pseudorandomness Game 2 of $\mathcal{F}$ that uses the reduced algebraic relation finder $\mathcal{B}$ from Algorithm 2 as offline algorithm and the online part $\mathcal{A}_{\mathsf{on}}$ of the basic algebraic attack Algorithm 4 as online algorithm.

If $m \geq 2^{2d-1} \cdot d^{d-1} \cdot n$, then the advantage of $\mathcal{A}$ in Game 2 is at least

$$\mathsf{adv}_{\mathcal{F}}^{\mathsf{PRG}}(\mathcal{A}) \geq (\#k_n)^{-\left\lceil 2^{\frac{d+1}{d-1}} \cdot \left(n^d/m\right)^{1/(d-1)} \right\rceil}, \tag{1.209}$$

and its total time complexity consists of $O(\log(m)^2 \cdot n^2)$ bit operations and

$$O\left(m^{3 \cdot \left\lceil 2^{\frac{d+1}{d-1}} \cdot (n^d/m)^{1/(d-1)} \right\rceil} \right) \tag{1.210}$$

arithmetic operations over k while its online complexity consists of

$$O\left(n \cdot m^{\left\lceil m^{2^{\frac{d+1}{d-1}} \cdot (n^d/m)^{1/(d-1)}} \right\rceil} \right) \tag{1.211}$$

arithmetic operations over k.

Corollary 32. *If $m \geq n^{1+e}$, the advantage of $\mathcal{A}$ is bounded by*

$$\mathsf{adv}_{\mathcal{F}}^{\mathsf{PRG}}(\mathcal{A}) \geq (\#k_n)^{-O(n^{1-e/(d-1)})}, \tag{1.212}$$

while its offline and online part perform

$$O\left(n^{3 \cdot (1+e) \cdot \left\lceil 2^{\frac{d+1}{d-1}} \cdot n^{1-e/(d-1)} \right\rceil} \right) \subset O(n^{n^{1-e/(d-1)}}) \tag{1.213}$$

and

$$O\left(n^{1+(1+e) \cdot \left\lceil 2^{\frac{d+1}{d-1}} \cdot n^{1-e/(d-1)} \right\rceil} \right) \subset O(n^{n^{1-e/(d-1)}}) \tag{1.214}$$

arithmetic operation over k, respectively.

Proof. The proof is analogous to the proof of Theorem 29 for the basic algebraic attack. Let k be a field of size q, draw $F \leftarrow \mathcal{F}$ and let $h \in k[Y]$ be the hint computed by $\mathcal{B}$ for $\mathcal{A}_{\mathsf{on}}$. Note that Proposition 25 implies the following properties for h

$$h \notin I_q \subset k[Y], \tag{1.215}$$

$$h(f_1, \ldots, f_m) \in I_q \subset k[Y], \tag{1.216}$$

$$\deg h \leq \left\lceil 2^{\frac{d+1}{d-1}} \cdot \left(n^d/m\right)^{1/(d-1)} \right\rceil, \tag{1.217}$$

where $f_1, \ldots, f_m \in k[X]$ are the polynomials that compute each output value of F.

Let (F, y, h) be the input for $\mathcal{A}_{\mathsf{on}}$. If $y = F(x)$ for some $x \in k^n$, we claim that $h(y)$ must vanish. Indeed, $h \circ F = h(f_1, \ldots, f_m)$ lies in $I_q = (X_1^q - X_1, \ldots, X_n^q - X_n)$ and can be written as

$$h \circ F = \sum_{i=1}^{n} g_i(X) \cdot (X_i^q - X_i) \tag{1.218}$$

for suitable polynomials $g_1, \ldots, g_n \in k[X]$. The field equations $X_1^q - X_1, \ldots, X_n^q - X_n$ vanish on each point $x \in k^n$, since the size of k is q. Hence, $h \circ F$ must also vanish on each $x \in k^n$. It follows that $\mathcal{A}$ always recognizes an image point $y = F(x)$ of F and outputs zero in these cases.

Now, let $y \leftarrow k^m$ be drawn uniformly at random. Because of Proposition 25, h is not zero modulo the field equations of k. Lemma 5 implies now

$$\Pr_{y \leftarrow k^m}[h(y) = 0] \leq 1 - q^{-\deg h}. \tag{1.219}$$

Hence, the probability that $\mathcal{A}$ refutes a truly random point of k^m is at least $q^{-\deg h}$.

It follows now for the advantage of $\mathcal{A}$

$$\mathsf{adv}_{\mathcal{F}}^{\mathsf{PRG}}(\mathcal{A}) \geq 1 + q^{-\deg h} - 1 = q^{-\deg h} \geq q^{-\left\lceil 2^{\frac{d+1}{d-1}} \cdot \left(\frac{n^d}{m}\right)^{1/(d-1)} \right\rceil}. \tag{1.220}$$

The bounds for the time complexities of $\mathcal{A}$ follow now from Proposition 25 and Theorem 29. □

1.3.3 Extended Algebraic Attack

The advantage of the adversary of Theorem 31 is unsatisfactory. In particular, in the case of lightweight binary PRGs—which are very popular in cryptography—it is desirable to have fast algorithms with significantly larger advantage. In this subsection, we will present a trade-off where we sacrifice a small portion of the complexity to drastically increase the advantage of the attack. Concretely, in the case of a poly-stretch PRG

$$\mathcal{F} : \{0,1\}^n \to \{0,1\}^{n^{1+e}} \tag{1.221}$$

of constant degree d, we will trade in the time complexity $n^{O(n^{1-e/(d-1)})}$ and the advantage $2^{-O(n^{1-e/(d-1)})}$ for the time complexity $n^{O(\log(n)^{1/(d-1)} \cdot n^{1-e/(d-1)})}$ and the advantage $1 - o(1)$.

The technique that we will present in the following works so well that we can guarantee a high advantage and a subexponential time complexity for constant-degree PRGs over any finite field. Given a PRG $\mathcal{F} : k^n \to k^m$, our idea is to enlarge the field k to an extension field $\overline{k} \supset k$ s.t. $\#\overline{k} \geq n$. According to Lemma 9, we can use Shoup's algorithm to find an appropriate extension field $\overline{k}$ of degree $r \leq \log(n)$, which costs us $O(\#k \cdot \log(n)^{4.1})$ arithmetic operations over k.

Now, given a generator $\zeta \in \overline{k} = k[\zeta]$ we know that the map

$$\begin{aligned} \psi : k^r &\longrightarrow \overline{k} && (1.222)\\ (b_1, \dots, b_r) &\longmapsto b_1 + \zeta \cdot b_2 + \ldots + \zeta^{r-1} \cdot b_r && (1.223) \end{aligned}$$

is a k-linear isomorphism of vector spaces. Note that ψ can be computed by $O(r)$ arithmetic operations over k and $\overline{k}$.

By abuse of notation, we will extend this map on larger vector spaces: let $\ell \in \mathbb{N}$, we extend ψ to

$$\begin{aligned} \psi : k^{\ell r} &\longrightarrow \overline{k}^{\ell} && (1.224)\\ \begin{pmatrix} b_1 \\ \vdots \\ b_{\ell r} \end{pmatrix} &\longmapsto \begin{pmatrix} \psi(b_1, \dots, b_r) \\ \psi(b_{r+1}, \dots, b_{2r}) \\ \vdots \\ \psi(b_{\ell r - r+1}, \dots, b_{\ell r}) \end{pmatrix} && (1.225) \end{aligned}$$

by applying it block-wise.

Given r polynomials $f_1, \dots, f_r \in k[X]^{\leq d}$, we can apply ψ coefficient-wise by setting

$$\psi(f_1, \dots, f_r) := f_1 + \zeta \cdot f_2 + \ldots + \zeta^{r-1} \cdot f_r \in \overline{k}[X]. \qquad (1.226)$$

Since ψ is a linear operation, the degree of $\psi(f_1, \dots, f_r)$ equals the maximum of the degrees $\deg f_1, \dots, \deg f_r$. It follows that we can extend ψ to a k-linear isomorphism

$$\begin{aligned} \psi : (k[X]^{\leq d})^r &\longrightarrow \overline{k}[X]^{\leq d} && (1.227)\\ (f_1, \dots, f_r) &\longmapsto f_1 + \zeta \cdot f_2 + \ldots + \zeta^{r-1} \cdot f_r && (1.228) \end{aligned}$$

for every $d \in \mathbb{N}$. Again, we can extend ψ block-wise on vectors of polynomials to get a k-linear isomorphism

$$\begin{aligned} \psi : (k[X]^{\leq d})^{\ell r} &\longrightarrow (\overline{k}[X]^{\leq d})^{\ell} \\ \begin{pmatrix} f_1 \\ \vdots \\ f_{\ell r} \end{pmatrix} &\longmapsto \begin{pmatrix} \psi(f_1, \dots, f_r) \\ \psi(f_{r+1}, \dots, f_{2r}) \\ \vdots \\ \psi(f_{\ell r - r+1}, \dots, f_{\ell r}). \end{pmatrix} \end{aligned}$$

These extensions on vectors of elements and vectors of polynomials are compatible with each other in the following way: for $f_1, \dots, f_{\ell r} \in k[X]$ and $x \in k^n$, we have

$$\begin{aligned}
&\psi(f_1, \ldots, f_{\ell r})(x) && (1.229)\\
=&(\psi(f_1, \ldots, f_r), \ldots, \psi(f_{\ell r-r+1}, \ldots, f_{\ell r}))(x) && (1.230)\\
=&(f_1 + \ldots + \zeta^{r-1} f_r, \ldots, f_{\ell r-r+1} + \ldots + \zeta^{r-1} f_{\ell r})(x) && (1.231)\\
=&(f_1(x) + \ldots + \zeta^{r-1} f_r(x), \ldots, f_{\ell r-r+1}(x) + \ldots + \zeta^{r-1} f_{\ell r}(x)) && (1.232)\\
=&(\psi(f_1(x), \ldots, f_r(x)), \ldots, \psi(f_{\ell r-r+1}(x), \ldots, f_{\ell r}(x))) && (1.233)\\
=&\psi(f_1(x), \ldots, f_{\ell r}(x)). && (1.234)
\end{aligned}$$

Let us formalize these observations:

Proposition 33. *Let $k \subseteq \overline{k}$ be a field extension of degree r. Let $\ell, d \in \mathbb{N}$. The k-linear isomorphism*

$$\begin{aligned}
\psi : k^r &\longrightarrow \overline{k} && (1.235)\\
(b_1, \ldots, b_r) &\longmapsto b_1 + \zeta \cdot b_2 + \ldots + \zeta^{r-1} \cdot b_r && (1.236)
\end{aligned}$$

extends block-wise to k-linear isomorphisms

$$\begin{aligned}
\psi : k^{\ell r} &\longrightarrow \overline{k}^{\ell} && (1.237)\\
(b_1, \ldots, b_{\ell r}) &\longmapsto (\psi(b_1, \ldots, b_r), \ldots, \psi(b_{\ell r-r+1}, \ldots, b_{\ell r})) && (1.238)
\end{aligned}$$

and

$$\begin{aligned}
\psi : (k[X]^{\leq d})^{\ell r} &\longrightarrow (\overline{k}[X]^{\leq d})^{\ell} && (1.239)\\
(f_1, \ldots, f_\ell) &\longmapsto (\psi(f_1, \ldots, f_r), \ldots, \psi(f_{\ell r-r+1}, \ldots, f_{\ell r})). && (1.240)
\end{aligned}$$

Further, we have for each degree-d map $F : k^n \to k^{\ell \cdot r}$ and $x \in k^n$

$$\psi(F)(x) = \psi(F(x)). \qquad (1.241)$$

Now, let $\mathcal{F} : k^n \to k^m$ be a PRG of degree d over a small field k (e.g. $\#k \in o(n)$). Our idea is to change the field over which $\mathcal{F}$ is cast. For this end, choose an extension field $\overline{k}$ of k of degree $r \in O(\log n)$ s.t. $\#\overline{k} \geq n$ and assume for simplicity that m is a multiple of r. Draw $F \leftarrow \mathcal{F}$ and apply ψ on it, i.e., consider the map

$$\begin{aligned}
\psi(F) : k^n &\longrightarrow \overline{k}^{m/r} && (1.242)\\
x &\longmapsto \psi(F)(x) = \psi(F(x)). && (1.243)
\end{aligned}$$

Now, $\psi(F)$ is of degree d over $\overline{k}$. We claim that the distribution $\mathcal{F}'$ that samples $F \leftarrow \mathcal{F}$ and outputs $\psi(F)$ is a PRG of type $k^n \to \overline{k}^{m/r}$ that is pseudorandom if $\mathcal{F}$ is pseudorandomness. Indeed, since k^m and $\overline{k}^{m/r}$ are isomorphic via ψ, it follows that ψ maps the uniform distribution over k^m to the uniform distribution over $\overline{k}^{m/r}$. Hence, if an adversary cannot distinguish between $F(x)$, $F \leftarrow \mathcal{F}, x \leftarrow k^n$, and $y \leftarrow k^m$, it also cannot distinguish between $\psi(F(x)) = \psi(F)(x)$ and $\psi(y)$, where the latter is uniformly distributed over $\overline{k}^{m/r}$. This gives us a cryptographic reduction, which shows that $\mathcal{F}' : k^n \to \overline{k}^{m/r}$ must be pseudorandom if $\mathcal{F} : k^n \to k^m$ is pseudorandom.

Now, if $m \in \omega(n \cdot r)$, we can apply the basic algebraic attack $\mathcal{A}$ from Algorithm 4 on $\mathcal{F}'$. Theorem 29 states that $\mathcal{A}$ has a high advantage of

$$\begin{aligned} \mathsf{adv}^{\mathsf{PRG}}_{\mathcal{F}'}(\mathcal{A}) &\geq 1 - O\left(\frac{\left(rn^d/m\right)^{1/(d-1)}}{\#\overline{k}}\right) && (1.244)\\ &\geq 1 - O\left(\frac{\left(rn^d/m\right)^{1/(d-1)}}{n}\right) && (1.245)\\ &\geq 1 - O\Big(\left(rn/m\right)^{1/(d-1)}\Big) \geq 1 - o(1) && (1.246) \end{aligned}$$

and a time complexity of

$$m^{O\left((rn^d/m)^{1/(d-1)}\right)} = \left(m^{O\left((n^d/m)^{1/(d-1)}\right)}\right)^{r^{1/(d-1)}}. \tag{1.247}$$

In comparison, the reduced attack from Theorem 31 has an advantage of

$$\geq 2^{-O((n^d/m)^{1/(d-1)})} \tag{1.248}$$

and a time complexity of

$$m^{O((n^d/m)^{1/(d-1)})}. \tag{1.249}$$

We see that we can increase the advantage of the algebraic attack from subexponentially small to high by dividing the stretch of the PRG by a logarithmic amount and increasing the time complexity of the algorithm by a logarithmic factor in the exponent. Let us formalize this attack:

Algorithm 5 (Extended Algebraic Attack)**.** Let k be a finite field and let $r : \mathbb{N} \to \mathbb{N}$. Also, let $\mathcal{F} : k^n \to k^m$ be a PRG of constant degree d and stretch $m > r(n) \cdot n$.

We will construct here an adversary $\mathcal{A} = (\mathcal{A}_{\mathsf{off}}, \mathcal{A}_{\mathsf{on}})$ for the pseudorandomness Game 2 of $\mathcal{F}$. $\mathcal{A}$ is parametrized by r. The offline part $\mathcal{A}_{\mathsf{off}}$ of $\mathcal{A}$ proceeds as follows:

Step 1: On input a map $F \leftarrow \mathcal{F}_n$ that is computed by polynomials $f_1, \ldots, f_{m(n)} \in k[X]$, $\mathcal{A}_{\mathsf{off}}$ uses Shoup's algorithm [Sho88] to compute a description of an extension field $\overline{k}$ of k of degree $r(n)$ together with a generator ζ. Let us denote this description by s.

Step 2: $\mathcal{A}_{\mathsf{off}}$ sets

$$m' := \left\lfloor \frac{m(n)}{r(n)} \right\rfloor, \tag{1.250}$$

and for $i \in [m']$ it computes

$$\begin{aligned} f'_i :=& \psi(f_{1+(i-1)\cdot r(n)}, \ldots, f_{i \cdot r(n)}) && (1.251)\\ =& f_{1+(i-1)\cdot r(n)} + \zeta \cdot f_{2+(i-1)\cdot r(n)} + \zeta^{r-1} \cdot f_{i \cdot r(n)} \in \overline{k}[X]. && (1.252) \end{aligned}$$

Step 3: Denote the algebraic relation finder from Algorithm 1 by $\mathcal{B}$, and denote by $\overline{k}[Y'] = \overline{k}[Y'_1, \ldots, Y'_{m'}]$ the polynomial ring over $\overline{k}$ in m' variables. $\mathcal{A}_{\mathsf{off}}$ evaluates $\mathcal{B}$ on $f'_1, \ldots, f'_m$ and receives an algebraic relation $h = \mathcal{B}(f'_1, \ldots, f'_m) \in \overline{k}[Y']$.

Step 4: $\mathcal{A}_{\mathsf{off}}$ outputs (s, h) as hint.

The online part $\mathcal{A}_{\mathsf{on}}$ of $\mathcal{A}$ is given by:

Step 1: On input the degree-d map $F : k^n \to k^{m(n)}$, a point $y \in k^m$ and the hint (s, h) from $\mathcal{A}_{\mathsf{off}}$, $\mathcal{A}_{\mathsf{on}}$ applies ψ on the first $r(n) \cdot m'$ coordinates of y, i.e.,

$$y' := \psi(y_1, \ldots, y_{m'}) = \begin{pmatrix} \psi(y_1, \ldots, y_{r(n)}) \\ \vdots \\ \psi(y_{m'-r(n)+1}, \ldots, y_{m'}) \end{pmatrix}. \tag{1.253}$$

Step 2: $\mathcal{A}_{\mathsf{on}}$ evaluates h on y', i.e., it evaluates

$$z := h(y'). \tag{1.254}$$

Step 3: If $z = 0$, then $\mathcal{A}_{\mathsf{on}}$ outputs 0, to signal that y is an output of F. Otherwise, $\mathcal{A}_{\mathsf{on}}$ outputs 1, which means that y is uniformly at random.

Theorem 34. *Let $\mathcal{F} : k^n \to k^m$ be a PRG of constant degree d over a finite field k and of stretch*

$$m \geq 2^{2d-1} \cdot d^{d-1} \cdot r(n) \cdot n \tag{1.255}$$

for a parameter $r : \mathbb{N} \to \mathbb{N}$.

The adversary $\mathcal{A}$ from Algorithm 5 has an advantage of

$$\mathsf{adv}_{\mathcal{F}}^{\mathsf{PRG}}(\mathcal{A}) \geq 1 - \frac{\left\lceil \left(\frac{r \cdot (2n)^d}{m-r} \right)^{\frac{1}{d-1}} \right\rceil}{\# k^r}, \tag{1.256}$$

a total time complexity of $O(\log(m)^2 \cdot n^2)$ bit operations and

$$O\left(r^{4.1} \cdot \# k + r^2 \cdot \left(\frac{m}{r} \right)^{3 \cdot \left\lceil 2^{\frac{d}{d-1}} \cdot \left(\frac{rn^d}{m-r} \right)^{1/(d-1)} \right\rceil} \right) \tag{1.257}$$

arithmetic operations over k and an online time complexity of

$$O\left(n \cdot r^2 \cdot \left(\frac{m}{r} \right)^{\left\lceil 2^{\frac{d}{d-1}} \cdot \left(\frac{rn^d}{m-r} \right)^{1/(d-1)} \right\rceil} \right) \tag{1.258}$$

arithmetic operations over k.

Corollary 35. *If $m \geq n^{1+e}$ for some constant $e > 0$ and $\# k \leq n$, then adversary $\mathcal{A}$ with $r(n) = \lceil \log n \rceil$ has a high advantage of $1 - o(1)$, a total time complexity of $O(\log(m)^2 \cdot n^2)$ bit operations and*

$$O\left(n^{(1+e) \cdot 3 \cdot \left\lceil 2^{\frac{d+1}{d-1}} \cdot \log(n)^{1/(d-1)} \cdot n^{1-e/(d-1)} \right\rceil} \right) \subset n^{O\left(\log(n)^{1/(d-1)} \cdot n^{1-e/(d-1)} \right)} \tag{1.259}$$

arithmetic operations over k, and an online time complexity of

$$O\left(n^{1+\left\lceil 2^{\frac{d+1}{d-1}}\cdot\log(n)^{1/(d-1)}\cdot n^{1-e/(d-1)}\right\rceil}\right) \subset n^{O\left(\log(n)^{1/(d-1)}\cdot n^{1-e/(d-1)}\right)} \tag{1.260}$$

arithmetic operations over k.

If $m \geq n^{1+e+e'}$ for constants $e > e' > 0$ and if $\#k \leq n$, then adversary $\mathcal{A}$ with $r(n) = \left\lceil n^{e'} \right\rceil$ has an overwhelming advantage of

$$1 - o\left(\frac{n}{2^{n^{e'}}}\right), \tag{1.261}$$

a total time complexity of $O(\log(m)^2 \cdot n^2)$ bit operations and

$$O\left(n^{(1+e)\cdot 3\cdot\left\lceil 2^{\frac{d+1}{d-1}}\cdot n^{1-e/(d-1)}\right\rceil}\right) \subset n^{O\left(n^{1-e/(d-1)}\right)} \tag{1.262}$$

arithmetic operations over k, and an online time complexity of

$$O\left(n^{1+\left\lceil 2^{\frac{d+1}{d-1}}\cdot n^{1-e/(d-1)}\right\rceil}\right) \subset n^{O\left(n^{1-e/(d-1)}\right)} \tag{1.263}$$

arithmetic operations over k.

Proof Theorem 34. Let us prove the bounds for the advantage, the offline and the online time complexity of $\mathcal{A}$ separately:

Advantage: Let $F : k^n \to k^m$ be of degree d, and let $f'_1, \ldots, f'_{m'} \in \overline{k}[X]$ be the polynomials computed by $\mathcal{A}_{\mathsf{on}}$ in Step 2, i.e.,

$$f'_i = \psi(f_{(i-1)\cdot r(n)+1}, \ldots, f_{i\cdot r(n)}). \tag{1.264}$$

Since $m' > 2^{2d-1} \cdot d^{d-1} \cdot n$, Corollary 18 guarantees that $\mathcal{B}$ in Step 3 outputs a polynomial $h \in \overline{k}[Y']$ s.t.

$$h \neq 0, \tag{1.265}$$

$$\deg h \leq \left\lceil \left(\frac{(2\cdot n)^d}{m'}\right)^{1/(d-1)} \right\rceil \tag{1.266}$$

$$h(f'_1, \ldots, f'_m) = 0. \tag{1.267}$$

Since $m' \geq \frac{m(n)-r(n)}{r(n)}$, we have for the degree of h

$$\deg h \leq \left\lceil \left(\frac{(2\cdot n)^d}{m'}\right)^{1/(d-1)} \right\rceil \leq \left\lceil \left(\frac{r(n)\cdot(2\cdot n)^d}{m(n)-r(n)}\right)^{1/(d-1)} \right\rceil. \tag{1.268}$$

Now, consider $\mathcal{A}_{\mathsf{on}}$. We claim that if $y \in k^m$ lies in the image of F, then $\mathcal{A}_{\mathsf{on}}$ will always output 0. And, if $y \leftarrow k^m$ is sampled uniformly at random, then $\mathcal{A}_{\mathsf{on}}$

will output 0 with probability $\leq \frac{\deg h}{\#\overline{k}}$. Let $y = F(x)$ for some $x \in k^n$. Then, we have for the coordinates of the vector $y' \in k^{m'}$ computed by $\mathcal{A}_{\mathsf{on}}$ in Step 1

$$\begin{aligned} y'_i =& \psi(y_{(i-1)\cdot r(n)+1}, \ldots, y_{i \cdot r(n)}) && (1.269)\\ =& \psi(f_{(i-1)\cdot r(n)+1}(x), \ldots, f_{i\cdot r(n)}(x)) && (1.270)\\ =& \psi(f_{(i-1)\cdot r(n)+1}, \ldots, f_{i\cdot r(n)})(x) = f'_i(x). && (1.271) \end{aligned}$$

In this case, we have in Step 3

$$h(y') = h(f'_1(x), \ldots, f'_{m'}(x)) = h(f'_1, \ldots, f'_{m'})(x) = 0(x) = 0. \tag{1.272}$$

Hence, $\mathcal{A}_{\mathsf{on}}$ will always output 0 if y lies in the image of F.

Now, let $y \leftarrow k^m$ be sampled uniformly at random. Since each $y'_i \in \overline{k}$ is computed by applying ψ on the i-th block of y, the vector y' is distributed uniformly at random over $\overline{k}^{m'}$, too. In particular, y' is independent of h, and the Schwartz-Zippel Lemma 2 implies

$$\Pr_{y' \leftarrow \overline{k}^{m'}}[h(y') = 0] \leq \frac{\deg h}{\#\overline{k}}. \tag{1.273}$$

The advantage of $\mathcal{A}$ is now given by the probability of $\mathcal{A}_{\mathsf{on}}$ outputting 0 if y lies in the image of F plus the probability of $\mathcal{A}_{\mathsf{on}}$ outputting 1 if y is distributed uniformly at random minus one. Hence, we have

$$\mathsf{adv}^{\mathsf{PRG}}_{\mathcal{F}}(\mathcal{A}) \geq 1 + 1 - \frac{\deg h}{\#\overline{k}} - 1 = 1 - \frac{\deg h}{\#\overline{k}}. \tag{1.274}$$

By inserting $\#\overline{k} = \#k^r$ and the bound for $\deg h$, the claim about the advantage of $\mathcal{A}$ follows.

Offline Time Complexity: The time complexity of $\mathcal{A}_{\mathsf{off}}$ is dominated by the number of its operations in Step 1 and Step 3. According to Lemma 9, the execution of Shoup's algorithm in Step 1 requires $O(\#k \cdot r^{4.1})$ arithmetic operations over k. Corollary 18 states that $\mathcal{B}$ in Step 3 makes $O(\log(m' + D)^2 \cdot D^2) \subseteq O(\log(m)^2 \cdot n^2)$ bit operations and $O(m'^{3D})$ arithmetic operations over $\overline{k}$ for

$$D(n) \leq \left\lceil \left(\frac{r \cdot (2n)^d}{m - r} \right)^{\frac{1}{d-1}} \right\rceil. \tag{1.275}$$

We can emulate each operation over $\overline{k}$ by $O(r^2)$ arithmetic operations over k, and hence get

$$O\left(r^{4.1} \cdot \#k + r^2 \cdot \left(\frac{m}{r}\right)^{3 \cdot \left\lceil 2^{\frac{d}{d-1}} \cdot \left(\frac{rn^d}{m-r}\right)^{1/(d-1)} \right\rceil} \right) \tag{1.276}$$

arithmetic operations over k.

Online Time Complexity: The online time complexity of $\mathcal{A}_{\mathsf{on}}$ is dominated by applying h on y' in Step 2. Evaluating h requires

$$O\Big(D \cdot \binom{m' + D}{D}\Big) \subseteq O\left(n \cdot \left(\frac{m}{r}\right)^{\left\lceil 2^{\frac{d}{d-1}} \cdot \left(\frac{rn^d}{m-r}\right)^{1/(d-1)} \right\rceil} \right) \tag{1.277}$$

arithmetic operations over $\overline{k}$, which yields

$$O\left(r^2 \cdot n \cdot \left(\frac{m}{r}\right)^{\left\lceil 2^{\frac{d}{d-1}} \cdot \left(\frac{rn^d}{m-r}\right)^{1/(d-1)} \right\rceil}\right) \tag{1.278}$$

arithmetic operations over k. □

Remark 3. Similarly to Theorem 29, we can also extend Algorithm 5 and Theorem 34 to the more general case of PRGs

$$\mathcal{F} : S^n \to T^n \tag{1.279}$$

of constant degree over some field k whose input and output elements are restricted to lie in subsets $S, T \subset k$. In such cases, the advantage of the basic algebraic attack from Algorithm 4 drops if the size of T is low. This problem can be fixed by utilizing Algorithm 5 with the parameter $r(n) = \left\lceil \frac{\log n}{\log \# T_n} \right\rceil$, which yields an adversary with high advantage.

1.4 Search-To-Decision Reduction

We will give here a generic search-to-decision reduction that can be implemented with our attacks from Section 1.3 to solve multivariate search problems in the average case. Our approach follows the usual search-to-decision reductions that are already known for Learning with Errors [Reg05] and Learning Parity with Noise [Pie12]. Given a pair $(F, F(x))$ for a polynomial map $F : k^n \to k^m$ and some $x \in k^n$, the idea would be to fix for each $i \in [n]$ and $z \in k$ the i-th variable in F to be z, i.e., we set

$$F_i(X_1, \ldots, X_{n-1}) := F(X_1, \ldots, X_{i-1}, z, X_i, \ldots, X_{n-1}). \tag{1.280}$$

Then, we give the pair $(F_i, F(x))$ to a PRG adversary $\mathcal{B}$ for polynomial maps. If $\mathcal{B}$ decides that $F(x)$ is only pseudorandom, we can assume that x_i must equal z. Otherwise, x_i does not equal z, and we continue to try another element of k. This leads to a search algorithm that executes the decision algorithm at most $n \cdot \# k$ times. However, note that this approach has two flaws:

1. Since we are invoking the PRG adversary $\mathcal{B}$ multiple times on the same source of randomness (F, x), we need that $\mathcal{B}$ has a high advantage.

2. The distribution of the target point $F(x)$ is not independent of the distribution F_i, hence $\mathcal{B}$ does not need to have a high advantage on input $(F_i, F_i(x))$. We solve this problem by manipulating F_i and $F_i(x)$ slightly to ensure that the distribution of the polynomial map and the target value that $\mathcal{B}$ receives are independent of each other.

In the case of random multivariate polynomial systems that contain a polynomial number of equations, the reduction presented here will yield a subexponential time algorithm with high success probability.

Let us first state the distributions of polynomial maps that are of interest for us:

Definition 14. Let k be a finite field and $n, m, d \in \mathbb{N}$. We denote by $\mathcal{MV}_{k,d}^{n,m}$ the uniform distribution over $(k[X_1, \ldots, X_n]^{\leq d})^m$. I.e., a map $F \leftarrow \mathcal{MV}_{k,d}^{n,m}$ consists of m polynomials of degree $\leq d$ over n variables whose coefficients are distributed uniformly and independently at random over k.

We can now state the corresponding search problems.

Definition 15 (Multivariate Search Problem)**.** Let $k = (k_n)_n$ be a family of fields and let $d \in \mathbb{N}$ be constant. Let $m : \mathbb{N} \to \mathbb{N}$ be a parameter.

For $F \leftarrow \mathcal{MV}_{k_n,d}^{n,m(n)}$ and $x \leftarrow k^n$, the **multivariate degree-d search problem** consists of finding x when given $(F, F(x))$.

Note that we state the multivariate search problem in the average-case, while in literature the multivariate quadratic (MQ) problem is usually stated in the worst-case [YDHTS15; BMSV22]. However, MQ is in the worst case **NP**-hard. Further, for most cryptographic applications the inversion hardness of *some* multivariate map is not sufficient. Hence, we think it is wiser to state the multivariate search problem in the average case. Assuming the hardness of this problem is equivalent to assuming the one-wayness of uniformly random polynomial maps of constant degree. One can also consider a decision version of the multivariate search problem where an adversary has to distinguish between $F(x)$ and a random vector $y \leftarrow k^m$. Assuming the hardness of this decision problem equals the assumption that random multivariate maps $F : k^n \to k^m$ are PRGs.

The distribution of the seeds given in Definition 15 is already of maximum hardness. I.e., for any distribution of seeds $\mathcal{X}$ over k^n, we can reduce the corresponding search problem $(F, F(x))$, for $F \leftarrow \mathcal{MV}_{k,d}^{n,m}$ and $x \leftarrow \mathcal{X}$, to the normal search problem $(F', F'(x'))$ with $F' \leftarrow \mathcal{MV}_{k,d}^{n,m}$ and $x' \leftarrow k^n$. Let us make this observations formal:

Lemma 36. *Let k be a finite field and $\mathcal{X}$ be any distribution over k^n. There is a reduction that maps instances $(F, F(x))$ for $F \leftarrow \mathcal{MV}_{k,d}^{n,m}$ and $x \leftarrow \mathcal{X}_n$ to instances $(F', F'(x'))$ for $F' \leftarrow \mathcal{MV}_{k,d}^{n,m}$ and $x' \leftarrow k^n$, s.t. each solution x' for $(F', F'(x'))$ yields a solution x for $(F, F(x))$. The reduction makes $O(n^3 + mn^d)$ operations over k to map $(F, F(x))$ to $(F', F'(x'))$ and $O(n^2)$ operations to map x' to x. The reduction has a failing probability of*

$$1 - \prod_{i=1}^{n} (1 - \#k_n^{-i}) \leq \frac{1}{\#k_n - 1}, \tag{1.281}$$

in which case it will directly abort.

Proof. The reduction $\mathcal{R}$ works as follows:

Step 1: On input $F : k^n \to k^m$ and $x \in k^n$, $\mathcal{R}$ samples

$$A \leftarrow k^{n \times k} \tag{1.282}$$

uniformly at random and checks the regularity of A by Gaussian elimination. If A is not invertible, then $\mathcal{R}$ aborts.

Step 2: The reduction computes $F' := F \circ A$, i.e., it sets for $i \in [m]$

$$f'_i(X) := f_i(A \cdot X) \tag{1.283}$$
$$= f_i(a_{1,1}X_1 + \ldots + a_{1,n}X_n, \ldots, a_{n,1}X_1 + \ldots + a_{n,n}X_n), \tag{1.284}$$

where $f_1, \ldots, f_m$ are the polynomials of F and $(a_{i,j})_{i,j\in[n]}$ are the entries of A.

Step 3: $\mathcal{R}$ outputs $(F', F(x))$.

Step 4: If $\mathcal{R}$ receives a solution x' for $(F', F(x))$, it computes and outputs

$$x := Ax'. \tag{1.285}$$

Let q be the size of k. The number of regular matrices in $k^{n\times n}$ is given by

$$(q^n - 1) \cdot (q^n - q) \cdots (q^n - q^{n-1}). \tag{1.286}$$

Hence, the probability that $\mathcal{R}$ aborts is $1 - \prod_{i=0}^{n-1}(1 - \#k_n^{i-n})$. Assume that A is invertible. We claim that the distribution of $(F \circ A, F(x))$ is identical to the distribution of $(F', F'(x'))$ for $F, F' \leftarrow \mathcal{MV}_{d,k}^{n,m}$, $x \leftarrow \mathcal{X}_n$ and $x' \leftarrow k^n$. Indeed, we have

$$(F \circ A, F(x)) = (F \circ A, F(AA^{-1}x)) = (F \circ A, F \circ A(A^{-1}x)). \tag{1.287}$$

The vector $A^{-1}x$ is distributed uniformly over k^n. Further, since A is of full rank, $F \circ A$ is distributed according to $\mathcal{MV}_{d,k}^{n,m}$. Lastly, the random variables $F \circ A$ and $A^{-1}x$ are independent of each other, since F is independent of x (F could be written as $G \circ A^{-1}$ for $G \leftarrow \mathcal{MV}_{d,k}^{n,m}$, which is independent of x).

If x' is a solution for $(F', F(x))$, then $A^{-1}x'$ is a solution for $(F, F(x))$, since

$$F(Ax') = F \circ A(x') = F'(x') = F(x). \qquad \square$$

Now, let $F : k^n \to k^m$ be of degree $d \in \mathbb{N}$ and let $y \in k^m$, $z \in k$ and $i \in [n]$. Denote by

$$F(0, \ldots, 0, z, 0, \ldots, 0) \tag{1.288}$$

the evaluation of F at the point whose i-th coordinate is z and whose all other coordinates are zero. Then, we set

$$D_i(F, z)(X_1, \ldots, X_{n-1}) := F(X_1, \ldots, X_{i-1}, z, X_i, \ldots, X_n) \tag{1.289}$$
$$- F(0, \ldots, 0, z, 0, \ldots, 0) + F(0), \tag{1.290}$$
$$E_i(y, F, z) := y - F(0, \ldots, 0, z, 0, \ldots, 0) + F(0). \tag{1.291}$$

Note that $D_i(F, z)$ is a polynomial map $k^{n-1} \to k^m$ of degree d and that $E_i(y, f, z)$ lies in k^m.

We have the following for D_i and E_i:

Lemma 37. *1. The point $x \in k^{n-1}$ is a solution for*

$$D_i(F, z)(X_1, \ldots, X_{n-1}) = E_i(y, F, z) \tag{1.292}$$

iff the point $(x_1, \ldots, x_{i-1}, z, x_i, \ldots, x_{n-1}) \in k^n$ is a solution for

$$F(X) = y. \tag{1.293}$$

2. Fix $x \in k^n$ and $z \in k$. If $z \neq x_i$, then the distribution of

$$(D_i(F, z), E_i(F(x), F, z)) \tag{1.294}$$

for $F \leftarrow \mathcal{MV}_{k,d}^{n,m}$ is identical to the distribution of

$$(F', y) \tag{1.295}$$

for $F' \leftarrow \mathcal{MV}_{k,d}^{n-1,m}$, $y \leftarrow k^m$.

3. For $F \leftarrow \mathcal{MV}_{k,d}^{n,m}$ and $x \leftarrow k^n$, the distribution

$$(D_i(F, x_i), E_i(F(x), F, x_i)) \tag{1.296}$$

is identical to the distribution

$$(F', F'(x')) \tag{1.297}$$

for $F' \leftarrow \mathcal{MV}_{k,d}^{n-1,m}$, $x' \leftarrow k^{n-1}$.

Proof. Note that each claim can be verified row-wise. Hence, it suffices to consider the case $m = 1$, i.e., $F = f \in k[X]$ is a polynomial of degree d over n variables. Further, without loss of generality, we assume that i equals n.

Draw $f \leftarrow k[X]^{\leq d}$ and decompose f

$$f(X_1, \dots, X_n) = g(X_1, \dots, X_n) + h(X_n) \tag{1.298}$$

into the sum of a multivariate polynomial $g \in k[X]$ and a univariate polynomial $h \in k[X_n]$ such that the monomials $X_n^1, \dots, X_n^d$ do not appear non-trivially in g (note that the absolute term stays in g). Note that the remaining coefficients of g and h are distributed uniformly and independently at random.

We then have for $x \in k^n$ and $y, z \in k$

$$\begin{aligned} f(0, \dots, 0, z, 0 \dots, 0) - f(0) &= h(z), && (1.299)\\ D_n(f, z)(X_1, \dots, X_{n-1}) &= g(X_1, \dots, X_{n-1}, z), && (1.300)\\ E_n(y, f, z) &= y - h(z), && (1.301)\\ E_n(f(x), f, x_n) &= g(x). && (1.302) \end{aligned}$$

We can now prove each claim separately:

1. For $x \in k^{n-1}, y, z \in k$ and $f \in k[X]^{\leq d}$, we have

$$\begin{aligned} & D_n(f, z)(x) = E_n(y, f, z) && (1.303)\\ \iff & g(x, z) = y - h(z) && (1.304)\\ \iff & g(x, z) + h(z) = y && (1.305)\\ \iff & f(x, z) = y. && (1.306) \end{aligned}$$

2. Fix $x \in k^n, z \in k$ s.t. $x_n \neq z$. Draw $f \leftarrow k[X]^{\leq d}$. Now, we have

$$\begin{aligned} D_n(f, z) &= g(X_1, \dots, X_{n-1}, z) && (1.307)\\ E_n(f(x), f, z) &= f(x) - h(z) = g(x_1, \dots, x_n) + h(x_n) - h(z). && (1.308) \end{aligned}$$

Since $x_n \neq z$, $h(x_n)-h(z)$ is uniformly distributed in k and independent of g. Now, each monomial in $k[X_1,\dots,X_n]^{\leq d}$ appears in g with a uniformly random coefficient that is independent of h. It follows that $D_n(f,x_n) = g(X_1,\dots,X_{n-1},z)$ is distributed uniformly in $k[X_1,\dots,X_{n-1}]^{\leq d}$. Further, $E_n(f(x),f,x_n) = g(x_1,\dots,x_n) + h(x_n) - h(z)$ is distributed uniformly in k and both distributions are independent of each other.

3. Draw $x \in k^n$ and $f \leftarrow k[X]^{\leq d}$. We have

$$D_n(f,x_n) = g(X_1,\dots,X_{n-1},x_n) \tag{1.309}$$
$$E_n(f(x),f,x_n) = f(x) - h(x_n) = g(x_1,\dots,x_n). \tag{1.310}$$

As before, $g(X_1,\dots,X_{n-1},x_n)$ is distributed uniformly at random in the space $k[X_1,\dots,X_{n-1}]^{\leq d}$ and independently of $x_1,\dots,x_{n-1}$. With $f' = g(X_1,\dots,X_{n-1},x_n)$ and $x' = (x_1,\dots,x_{n-1})$, the distribution

$$(D_n(f,x_n),E_n(f(x),f,x_n)) \tag{1.311}$$

is identical to the distribution $(f',f'(x'))$. □

Algorithm 6 (Multivariate Search Algorithm)**.** Let $d \in \mathbb{N}$ be constant and let k be a field.

We will describe here an algorithm $\mathcal{A}$ for the multivariate search problem of Definition 15. The algorithm will make black-box usage of an adversary $\mathcal{D}$ for algebraic PRGs over k of degree d. $\mathcal{A}$ proceeds as follows:

Step 1: $\mathcal{A}$ receives as input a polynomial map $F : k^n \to k^m$ of degree d and a point $y \in k^m$. For each $i \in [n]$, $\mathcal{A}$ creates an empty set $L_i = \emptyset$.

Step 2: For each $i \in [n]$ and $z \in k$, $\mathcal{A}$ does the following subroutine:

Step 1: $\mathcal{A}$ computes

$$F' := D_i(F,z), \tag{1.312}$$
$$y' := E_i(y,F,z) \tag{1.313}$$

Step 2: $\mathcal{A}$ queries $\mathcal{D}$ on (F',y'). If $\mathcal{D}$ decides that y' is a pseudorandom output value of F', then $\mathcal{A}$ adds the element z to L_i.

Step 3: If there is a list L_i that does not contain exactly one element, then $\mathcal{A}$ aborts.

Step 4: Denote the element of each L_i by x_i. $\mathcal{A}$ outputs the point $(x_1,\dots,x_n) \in k^n$.

Theorem 38. *Let k be a finite field, let $d \in \mathbb{N}$ be constant and let $m : \mathbb{N} \to \mathbb{N}$ be a parameter.*

Let $\mathcal{A}$ be the solver for the multivariate degree-d search problem from Algorithm 6 with black-box access to an algebraic PRG adversary $\mathcal{D}$. Then, $\mathcal{A}$ makes $\Theta(\#k \cdot n^{d+1})$ arithmetic operations over k on its own and $\#k \cdot n$ calls of $\mathcal{D}$.

For the success probability of $\mathcal{A}$, we have

$$\Pr_{\substack{F \leftarrow \mathcal{MV}_{k,d}^{n,m} \\ x \leftarrow k^n}} \geq 1 - \#k \cdot n \cdot \left(1 - \mathsf{adv}^{\mathsf{PRG}}_{\mathcal{MV}_{k,d}^{n-1,m}}(\mathcal{D})\right) \tag{1.314}$$

where $\mathsf{adv}^{\mathsf{PRG}}_{\mathcal{MV}_{k,d}^{n-1,m}}(\mathcal{D})$ *is the advantage of $\mathcal{D}$ against the PRG*

$$\mathcal{MV}_{d,k}^{n-1,m} : k^{n-1} \longrightarrow k^m. \tag{1.315}$$

Corollary 39. *Set $r(n) := 1 + \left\lceil 2 \cdot \frac{\log(n)}{\log(\#k_n)} \right\rceil$ and let $m \in \omega(rn)$. If we instantiate the solver $\mathcal{A}$ from Algorithm 6 with the PRG adversary $\mathcal{D}$ from Algorithm 5 with parameter r, then $\mathcal{A}$ has a high success probability of $1 - o(1)$. Further, it can be implemented by making $O(\log(m^2) \cdot n^2)$ bit operations and*

$$O\left(\#k \cdot n \cdot r^2 \cdot \left(\frac{m}{r}\right)^{3 \cdot \left\lceil 2^{\frac{d}{d-1}} \cdot \left(\frac{rn^d}{m-r}\right)^{1/(d-1)} \right\rceil}\right) \tag{1.316}$$

arithmetic operations over k.

If $m \geq n^{1+e}$ for a constant $e > 0$, then the time complexity of $\mathcal{A}$ instantiated with $\mathcal{D}$ lies in

$$O\left(\#k \cdot n^{1+(1+e)\cdot 3 \cdot \left\lceil 2^{\frac{d+1}{d-1}} \cdot \log(n)^{1/(d-1)} \cdot n^{1-e/(d-1)} \right\rceil}\right) \subset \#k \cdot n^{O(\log(n)^{1/(d-1)} \cdot n^{1-e/(d-1)})}.$$

Proof. For the advantage of $\mathcal{D}$, we have

$$\begin{aligned} \mathsf{adv}^{\mathsf{PRG}}_{\mathcal{MV}^{n-1,m}_{k,d}} \geq & 1 - \frac{\left\lceil \left(\frac{r \cdot 2^d \cdot (n-1)^d}{m(n) - r(n)}\right)^{1/(d-1)} \right\rceil}{\#k_n^{r(n)}} & (1.317) \\ \geq & 1 - O\left(\frac{\left(r \cdot n^d / m\right)^{1/(d-1)}}{\#k_n \cdot n^2}\right) & (1.318) \\ \geq & 1 - o\left(\frac{1}{\#k_n \cdot n}\right). & (1.319) \end{aligned}$$

Hence, the claim for the advantage of $\mathcal{A}$ follows. Now, $\mathcal{D}$ computes $D(n-1)$, performs Shoup's algorithm to find an extension field of degree r and then performs

$$O\left(r^2 \cdot (m/r)^{3 \cdot \left\lceil 2^{\frac{d}{d-1}} \cdot \left(\frac{r\dot{(}n-1)^d}{m-r}\right)^{1/(d-1)} \right\rceil}\right) \tag{1.320}$$

arithmetic operations over k. The calculation of $D(n-1)$ and Shoup's algorithm can be excluded from the subroutine of $\mathcal{A}$. Hence, we get only $O(\log(m^2) \cdot n^2)$ bit operations, while the complexity of Shoup's algorithm is dominated by

$$\#k \cdot n \cdot O\left(r^2 \cdot (m/r)^{3 \cdot \left\lceil 2^{\frac{d}{d-1}} \cdot \left(\frac{rn^d}{m-r}\right)^{1/(d-1)} \right\rceil}\right). \qquad \square$$

Proof Theorem 38. The bounds for the time complexity of $\mathcal{A}$ are easy to see.

To prove the lower bound for the success probability of $\mathcal{A}$, note that $\mathcal{A}$ outputs x iff each call of $\mathcal{D}$ is successful. Hence, fix $i \in [n]$, $z \in k$ and consider the corresponding subroutine. If we have $z \neq x_i$, then Lemma 37 implies that the pair $F' = D_i(F, z)$ and $y' = E_i(y, F, z)$ is distributed according to $\mathcal{MV}^{n-1,m}_{d,k}$ and the uniform distribution over k^m. Hence, $\mathcal{D}$ must output in this case 1 with probability

$$\Pr_{\substack{F' \leftarrow \mathcal{MV}^{n-1,m}_{d,k}, \\ y' \leftarrow k^m}} [\mathcal{D}(F', y') = 1] \geq 1 - \mathsf{adv}^{\mathsf{PRG}}_{\mathcal{MV}^{n-1,m}_{d,k}}(\mathcal{D}). \tag{1.321}$$

Otherwise, if $z = x_i$, then F' is distributed according to $\mathcal{MV}_{d,k}^{n-1,m}$, and y' is distributed according to $F'(x')$ for $x' \leftarrow k^{n-1}$. In this case, we have

$$\Pr_{\substack{F' \leftarrow \mathcal{MV}_{d,k}^{n-1,m}, \\ x' \leftarrow k^{n-1}}} [\mathcal{D}(F', F'(x')) = 0] \geq 1 - \mathsf{adv}^{\mathsf{PRG}}_{\mathcal{MV}_{d,k}^{n-1,m}}(\mathcal{D}). \tag{1.322}$$

It follows that the probability that $\mathcal{D}$ judges correctly in each case is bounded by

$$\geq 1 - \#k \cdot n \cdot (1 - \mathsf{adv}^{\mathsf{PRG}}_{\mathcal{MV}_{d,k}^{n-1,m}}(\mathcal{D})). \qquad \square$$

We want to close this section by pointing out that, if the search-to-decision reduction $\mathcal{A}$ from Algorithm 6 is instantiated with a PRG adversary $\mathcal{D}$ from Section 1.3, then we can give even better guarantees for the success probability of $\mathcal{A}$, since $\mathcal{D}$ does not depend on the randomness of the function $F : k^n \to k^m$ or the seed x and has even in the worst-case a high advantage. For this end, let us define when a distribution of random polynomial maps has sufficient randomness for our search-to-decision reduction:

Definition 16. Let k be a field and $n, m, d \in \mathbb{N}$. We will say that a distribution $\mathcal{F}$ over $(k[X_1, \ldots, X_n]^{\leq d})^m$ has **linear core** if there is a distribution $\mathcal{L}$ s.t. $F \leftarrow \mathcal{F}$ is equally distributed to

$$G(X) + A \cdot (X_1, \ldots, X_n) \tag{1.323}$$

for $G \leftarrow \mathcal{L}$ and $A \leftarrow k^{m \times n}$.

Lemma 40. *Let $\mathcal{F}$ be a distribution over $(k[X_1, \ldots, X_n]^{\leq d})^m$ with linear core and let $r(n) = 1 + \left\lceil 2 \cdot \frac{\log(n)}{\log(\#k_n)} \right\rceil$, $m \in \omega(rn)$. Let $\mathcal{A}$ be the search-to-decision reduction from Algorithm 6 instantiated with the PRG adversary from Algorithm 5 with parameter $r(n) = 1 + \left\lceil 2 \cdot \frac{\log(n)}{\log(\#k_n)} \right\rceil$. We have for each $x \in k^n$*

$$\Pr_{F \leftarrow \mathcal{F}_n} [\mathcal{A}(F, F(x)) = x] \geq 1 - o(1). \tag{1.324}$$

Proof. Let $\mathcal{D}$ be the PRG adversary from Algorithm 5. Then, $\mathcal{D}$ will always output 0 on input $(F, F(x))$. Further, the bound that we have proven for the advantage of $\mathcal{D}$ on input (F', y') only depends on the randomness of y' and is independent of the distribution of F'.

Hence, it suffices to show that y' is uniformly at random and independent of F' in the subroutine of $\mathcal{A}$. Since $\mathcal{F}$ has a linear core, there is a distribution $\mathcal{L}$ s.t. $F \leftarrow \mathcal{F}_n$ is distributed as

$$F(X) = G(X) + A \cdot (X_1, \ldots, X_n) \tag{1.325}$$

for $G \leftarrow \mathcal{L}_n$ and $A \leftarrow k^{m \times n}$. Denote the columns of A by $a_1, \ldots, a_n$. Let $i \in [n]$ and $z \neq x_i$, and assume—without loss of generality—that $i = n$. We have

$$\begin{aligned} F' =& D_n(F, z) && (1.326)\\ =& F(X_1, \ldots, X_{n-1}, z) - F(0, \ldots, 0, z) + F(0) && (1.327)\\ =& G(X_1, \ldots, X_{n-1}, z) + A \cdot (X_1, \ldots, X_{n-1}, z) && (1.328)\\ & - G(0, \ldots, 0, z) - A \cdot (0, \ldots, 0, z) + G(0) && (1.329)\\ =& D_n(G, z) - \sum_{i=1}^{n-1} X_i \cdot a_i && (1.330) \end{aligned}$$

and

$$\begin{aligned} y' =& E_n(F(x), F, z) && (1.331)\\ =& F(x) - F(0, \dots, 0, z) + F(0) && (1.332)\\ =& G(x) + A \cdot x - G(0, \dots, 0, z) - A \cdot (0, \dots, 0, z) + G(0) && (1.333)\\ =& E_n(G(x), G, z) + (z - x_n) \cdot a_n. && (1.334) \end{aligned}$$

Now, a_n is a uniformly random vector in k^m that is independent of G and $a_1, \dots, a_{n-1}$. Hence, in the subroutine of $\mathcal{A}$ for $i \in [n]$ and $z \neq x_i$, the value y' is distributed uniformly at random and independent of F'. □

Note that the time complexity of Algorithm 6 depends linearly on the size of the field k (resp. the size of the set S of possible entries of the seed x). If k is of exponential size, then the time complexity of Algorithm 6 is not subexponential, any more. A possible solution in this case might be to decompose a seed value x_i into a sum

$$x_i = \sum_{j=1}^{\lceil \log q \rceil} \zeta_j \cdot b_{i,j} \qquad (1.335)$$

for bits $b_{i,1}, \dots, b_{i,\lceil \log q \rceil} \in \{0, 1\}$ and fixed field values $\zeta_1, \dots, \zeta_{\lceil \log q \rceil} \in k$, where $q = \#k$. This would allow one to repeatedly fix one of the bits and ask the PRG adversary Algorithm 4 for solvability of the corresponding equation system. This would yield a search algorithm with time complexity $\log(q) \cdot n^{O((n^d/m)^{1/(d-1)})}$ over large fields. However, it is not clear how to prove a high success probability for this algorithm. Note that in Algorithm 6 we move a part of the randomness of $F \leftarrow \mathcal{F}$ to the right-hand side of the equation $F(x) = y$ to guarantee that y' is distributed uniformly at random and independent of F'. When using the bit-decomposition of Eq. (1.335), this argument does not hold, any more. Hence, we ask here the following question:

Question 4. *Let k be a field of size $\#k \geq 2^n$ and let $m \geq n^{1+e}$ for some $e > 0$ constant. For a constant $d > 1$, is there an algorithm that can solve the multivariate degree-d search problem of Definition 15 with time complexity in*

$$\log(\#k) \cdot n^{O(n^{1-e/(d-1)})} \qquad (1.336)$$

and high success probability?

1.5 On Macaulay Matrix-Based Attacks

We close this chapter by comparing the distinguishing and search algorithms of Sections 1.3 and 1.4 with Macaulay matrix-based algorithms, which predate the algorithms of this work. While Macaulay matrix-based algorithms are widely used in the cryptanalysis of multivariate cryptosystems [Fau99; CKPS00; Fau02; YC04; YC05; DBMMW08; MMDB08; TW10; Alb10; CCNY12], there are not a lot of formal guarantees known for those algorithms. In fact, most of their performance estimation is based on heuristics. We will show here that Macaulay matrix-based algorithms and our algorithms based on algebraic relations stand in a direct correspondence. This will allow us to derive upper bounds for the time complexity and advantage of Macaulay matrix-based algorithms.

For an overview of bounds when computing Groebner bases, we refer the reader to the excellent overviews of Caminata and Gorla [CG21; CG23] and to the important bounds given by Möller and Mora [MM84], Huynh [Huy86], and Dubé [Dub90].

Let us first introduce the notion of Macaulay matrices [Mac02; Mac16]:

Definition 17. Let $n, m \in \mathbb{N}$ and let k be a field. Let $F : k^n \to k^m$ be a polynomial map of polynomials $f_1, \ldots, f_m \in k[X_1, \ldots, X_n]^{\leq d}$ and let $y \in k^m$.

We define the **Macaulay map** with respect to the equation system $F(X) = y$ as the following linear map

$$\mu : (k[X])^m \longrightarrow k[X] \tag{1.337}$$

$$(g_1, \ldots, g_m) \longmapsto g_1 \cdot (f_1 - y_1) + \ldots + g_m \cdot (f_m - y_m). \tag{1.338}$$

For $L \in \mathbb{N}$, we define the Macaulay map up to degree L by the following restriction of μ

$$\mu^{\leq L} : \left(k[X]^{\leq L-d}\right)^m \longrightarrow k[X]^{\leq L}, \tag{1.339}$$

where we set $k[X]^{\leq a} := 0$ whenever $a < 0$.

The **Macaulay matrix** M_L up to degree L is given as the transpose of the matrix representation of $\mu^{\leq L}$ with respect to the monomial basis of $k[X]^{\leq L}$.

If the functions $f_1, \ldots, f_m$ are of degree d and $L > d$, then their Macaulay matrix M_L is of shape $m \cdot \binom{n+L-d}{L-d} \times \binom{n+L}{L}$. This means that each block of $\binom{n+L-d}{L-d}$ rows represents the coefficients of a polynomial $g_i \in k[X]^{\leq L-d}$, while each column represents a monomial in $k[X]^{\leq L}$. Usually, one requires that the columns of the Macaulay matrix are sorted according to some monomial ordering. However, since we are not interested in Groebner bases, monomial orderings will not matter in our case.

Note that the image of μ is exactly the ideal

$$(f_1 - y_1, \ldots, f_m - y_m) \subseteq k[X] \tag{1.340}$$

generated by the equations of $F(X) = y$. Hilbert's Nullstellensatz implies now that $F(X) = y$ has a solution over the algebraic closure of k iff 1 does not lie in the image of μ. Further, $F(X) = y$ has exactly one solution x over the algebraic closure of k iff the radical of $(f_1 - y_1, \ldots, f_m - y_m)$ equals $(X_1 - x_1, \ldots, X_n - x_n) \subset k[X]$.

Based on these observations, let us give prototype algorithms for deciding the satisfiability and solving polynomial equation systems:

Algorithm 7 (Macaulay Matrix-Based Decider)**.** The algorithm $\mathcal{M}$ receives as input a list of polynomials $f_1, \ldots, f_m \in k[X]$ of degree d and a vector of values $y \in k^m$. It tries to decide if the equation system $F(x) = y$ is satisfiable. Additionally, $\mathcal{M}$ receives a control parameter $D \in \mathbb{N}$.

Step 1: On input $f_1, \ldots, f_m \in k[X]$, $y \in k^m$ and $D \in \mathbb{N}$, $\mathcal{M}$ repeats the following subroutine for $L = 1, \ldots, D$:

Step 1: $\mathcal{M}$ computes the Macaulay matrix $M_L \in k^{m\cdot\binom{n+L-d}{L-d}\times\binom{n+L}{L}}$ with respect to the equation system

$$f_1(X) = y_1, \dots, f_m(X) = y_m. \tag{1.341}$$

Step 2: $\mathcal{M}$ uses a linear equation solver to check if 1 lies in the row span

$$k^{m\cdot\binom{n+L-d}{L-d}} \cdot M_L \tag{1.342}$$

of M_L. If 1 lies in the row span of M_L, then $\mathcal{M}$ outputs 1 and terminates.

Step 2: If after D iterations $\mathcal{M}$ did not terminate, it outputs 0 and terminates.

Note that 1 lies in the row span of M_L iff 1 lies in the image of $\mu^{\leq L}$, which implies the insatisfiability of $F(X) = y$. Hence, $\mathcal{M}$ only outputs 1 iff it can extract a witness for the non-satisfiability of $F(X) = y$. However, if $F(X) = y$ is satisfiable, then $\mathcal{M}$ will never find a witness for its insatisfiability. Hence, we need to give the additional control parameter D, which tells $\mathcal{M}$ to assume the satisfiability of $F(X) = y$ if no contradictions up to degree D have shown up. Lazard [Laz83] showed that it suffices to take $D = (n + 1)(d - 1) + 1$, which would lead to an exponential time algorithm. Further, note that we did not specify the linear equation solver used by $\mathcal{M}$. In general, the sparse linear solver of Coppersmith [Cop94] is used. We will discuss this in more detail in Section 1.5.2.

Let us introduce a prototype solving algorithm for $F(X) = y$.

Algorithm 8 (Macaulay Matrix-Based Solver)**.** The algorithm $\mathcal{M}^{\mathsf{solv}}$ receives as input a list of polynomials $f_1, \dots, f_m \in k[X]$ of degree d and a vector of values $y \in k^m$. Additionally, $\mathcal{M}^{\mathsf{solv}}$ receives a control parameter $D \in \mathbb{N}$. It tries to compute a vector $x \in k^n$ s.t. $F(x) = y$.

Step 1: On input $f_1, \dots, f_m \in k[X]$, $y \in k^m$ and $D \in \mathbb{N}$, $\mathcal{M}^{\mathsf{solv}}$ repeats the following subroutine for $L = 1, \dots, D$:

Step 1: $\mathcal{M}^{\mathsf{solv}}$ computes the Macaulay matrix $M_L \in k^{m\cdot\binom{n+L-d}{L-d}\times\binom{n+L}{L}}$ with respect to the equation system

$$f_1(X) = y_1, \dots, f_m(X) = y_m. \tag{1.343}$$

Step 2: $\mathcal{M}^{\mathsf{solv}}$ uses a linear equation solver to check if the row span of M_L contains a polynomial of shape

$$X_i - z \tag{1.344}$$

for some $i \in [n]$. If so it sets $x_i := z$.

Step 3: If each $x_1, \dots, x_n$ is set at this point, then $\mathcal{M}^{\mathsf{solv}}$ terminates and outputs

$$x := (x_1, \dots, x_n) \in k^n. \tag{1.345}$$

Step 2: If after D iterations $\mathcal{M}^{\mathsf{solv}}$ did not terminate, it aborts.

In general, we cannot guarantee the correctness of $\mathcal{M}^{\mathsf{solv}}$. It $F(X) = y$ is unsatisfiable, then $\mathcal{M}^{\mathsf{solv}}$ may output an incorrect solution x instead of aborting. Further, even if $F(X) = y$ has a unique solution, $\mathcal{M}^{\mathsf{solv}}$ will not find it if the ideal $(f_1(X) - y_1, \dots, f_m(X) - y_m)$ is not radical.

We will discuss the time complexity of the distinguishing algorithms in Section 1.5.2, where we will compare it directly with the attack algorithms of Section 1.3. Before that, we will give upper bounds for both prototype Macaulay matrix-based algorithms in Section 1.5.1.

1.5.1 Upper Bounds for Macaulay Matrices

We will show in this subsection that the upper bounds from Theorems 29, 31 and 34 carry over to Macaulay matrix-based distinguishing algorithms:

Theorem 41. *Let k be a finite field of size $q = q(n)$ and let $\mathcal{F} : k^n \to k^m$ be a PRG of constant degree $d \in \mathbb{N}$ and stretch*

$$m(n) \geq 2^{2d-1} \cdot d^{d-1} \cdot n. \tag{1.346}$$

Further, let $D : \mathbb{N} \to \mathbb{N}$ be a parameter.

Let $\mathcal{M}$ be the distinguisher from Algorithm 7, parametrized with $d \cdot D(n)$. We consider $\mathcal{M}$ as an adversary in the pseudorandomness Game 2 of $\mathcal{F}$ without an offline algorithm.

We have the following for $n \geq 2d$:

1. *For $D(n) \geq \left\lceil 2^{\frac{d}{d-1}} \cdot \left(\frac{n^d}{m(n)}\right)^{1/(d-1)} \right\rceil$, we have for the advantage of $\mathcal{M}$ in Game 2*

$$\mathsf{adv}_{\mathcal{F}}^{\mathsf{PRG}}(\mathcal{M}) \geq 1 - \frac{\left\lceil 2^{\frac{d}{d-1}} \cdot \left(\frac{n^d}{m(n)}\right)^{1/(d-1)} \right\rceil}{q(n)}. \tag{1.347}$$

2. *For $r : \mathbb{N} \to \mathbb{N}$, if we have $m(n) \geq 2^{2d-1} \cdot d^{d-1} \cdot r(n) \cdot n$ and $D(n) \geq \left\lceil 2^{\frac{d}{d-1}} \cdot \left(\frac{r(n) \cdot n^d}{m(n) - r(n)}\right)^{\frac{1}{d-1}} \right\rceil$, then*

$$\mathsf{adv}_{\mathcal{F}}^{\mathsf{PRG}}(\mathcal{M}) \geq 1 - \frac{\left\lceil 2^{\frac{d}{d-1}} \cdot \left(\frac{r(n) \cdot n^d}{m(n) - r(n)}\right)^{\frac{1}{d-1}} \right\rceil}{q(n)^{r(n)}}. \tag{1.348}$$

3. *Let $D(n) \geq \left\lceil 2^{\frac{d+1}{d-1}} \cdot \left(\frac{n^d}{m(n)}\right)^{1/(d-1)} \right\rceil$, and consider now the distinguisher $\mathcal{M}$ from Algorithm 7 that receives in Game 2 as input the polynomials $f_1, \dots, f_m, X_1^q - X_1, \dots, X_n^q - X_n$ and the values $y_1, \dots, y_m, 0, \dots, 0$, i.e.,*

$\mathcal{M}$ has now to decide the satisfiability of

$$
\begin{aligned}
f_1(X_1, \ldots, X_n) &= y_1, && (1.349)\\
&\vdots && (1.350)\\
f_m(X_1, \ldots, X_n) &= y_m, && (1.351)\\
X_1^q - X_1 &= 0, && (1.352)\\
&\vdots && (1.353)\\
X_n^q - X_n &= 0. && (1.354)
\end{aligned}
$$

Then, we have for the advantage of $\mathcal{M}$ in Game 2

$$\mathsf{adv}_{\mathcal{F}}^{\mathsf{PRG}}(\mathcal{M}) \geq 1 - q(n)^{-\left\lceil 2^{\frac{d+1}{d-1}} \cdot \left(\frac{n^d}{m(n)}\right)^{1/(d-1)} \right\rceil}. \tag{1.355}$$

In general, one can show that the Macaulay matrix-based distinguishing Algorithm 7 terminates (with non-trivial resp. high advantage) after D iterations where

$$D := \min\left\{L \in \mathbb{N} \;\middle|\; \ker \phi^{\leq L} > 0\right\} \tag{1.356}$$

is the smallest degree such that the kernel of the corresponding dual morphism is non-trivial.

Analogously, we can transfer the guarantees for our search-to-decision reduction from Theorem 38 to the Macaulay matrix-based solving Algorithm 8:

Theorem 42. *Let k be a field of size q and let $\mathcal{F}$ be a distribution of polynomial maps $k^n \to k^m$ of degree d and stretch $m \in \omega(rn)$. Assume that $\mathcal{F}$ has a linear core in the sense of Definition 16. Set $r(n) := 1 + \left\lceil 2 \cdot \frac{\log n}{\log q} \right\rceil$ and*

$$D(n) := \left\lceil 2^{\frac{d}{d-1}} \cdot \left(\frac{r(n) \cdot n^d}{m(n) - r(n)}\right)^{1/(d-1)} \right\rceil. \tag{1.357}$$

Denote by $\mathcal{M}^{\mathsf{solv}}$ the search algorithm from Algorithm 8 instantiated with parameter $D(n) + q - 1$.

We have for each $x \in k^n$

$$\Pr_{F \leftarrow \mathcal{F}_n}\left[\mathcal{M}^{\mathsf{solv}}(F, X_1^q - X_1, \ldots, X_n^q - X_n, F(x), 0, \ldots, 0) = x\right] \geq 1 - o(1).$$

Both theorems imply that Macaulay matrix-based algorithms can attack with high advantage a PRG resp. a OWF $\mathcal{F} : k^n \to k^m$ of degree d and stretch $m \geq n^{1+e}$ by performing $n^{O(n^{1-e/(d-1)})}$ iterations, if k is large enough, resp. $n^{O(\log(n)^{1/d-1} \cdot n^{1-e/(d-1)})}$ iterations for small fields k. To prove both theorems, we introduce the regular Macaulay map, which acts as a bridge between the dual morphism and the normal Macaulay map.

Definition 18. Let $k[X, Y]$ be the polynomial ring in n variables $X_1, \ldots, X_n$ of degree 1 and m additional variables $Y_1, \ldots, Y_m$ of degree d. I.e., the degree of a monomial $X_1^{a_1} \cdots X_n^{a_n} \cdot Y_1^{b_1} \cdots Y_m^{b_m}$ is given by

$$\deg(X_1^{a_1} \cdots X_n^{a_n} \cdot Y_1^{b_1} \cdots Y_m^{b_m}) = a_1 + \ldots + a_n + d \cdot (b_1 + \ldots b_m). \tag{1.358}$$

The grading $k[X,Y] = \bigoplus_{\ell=0}^{\infty} k[X,Y]^{\ell}$ of $k[X,Y]$ is given with respect to the degrees of $X_1,\ldots,X_n$ and $Y_1,\ldots,Y_m$.

For a list of polynomials $f_1,\ldots,f_m \in k[X]$ of degree d, we define the **regular Macaulay map** by

$$\mu_{\mathsf{reg}} : (k[X,Y])^m \longrightarrow k[X,Y] \tag{1.359}$$

$$(g_1(X,Y),\ldots,g_m(X,Y)) \longrightarrow \sum_{i=1}^{m} g_i(X,Y)\cdot(f_i(X) - Y_i). \tag{1.360}$$

Up to degree $L \in \mathbb{N}$, the regular Macaulay map is given by its restriction

$$\mu_{\mathsf{reg}}^{\leq L} : \left(k[X,Y]^{\leq L-d}\right)^m \longrightarrow k[X,Y]^{\leq L}, \tag{1.361}$$

where we set $k[X,Y]^{\leq a} := 0$ for $a < 0$.

We call the map μ_{reg} the *regular* Macaulay map, since the polynomials $f_1(X) - Y_1,\ldots,f_m(X) - Y_m$ form a regular sequence in $k[X,Y]$. (This implies that the kernel of μ_{reg} only contains trivial syzygies.)

Between the regular Macaulay map μ_{reg} and the dual morphism ϕ of $f_1,\ldots,f_m$, the following interplay exists:

Lemma 43. *Let $m,n,d \in \mathbb{N}$. Let $f_1,\ldots,f_m \in k[X]^{\leq d}$ where k is a field. Denote by $\operatorname{Img}\mu_{\mathsf{reg}}$ the image resp. range of μ_{reg}, then we have*

$$\operatorname{Img}\mu_{\mathsf{reg}} \cap k[Y] = \ker\phi, \tag{1.362}$$

and for $L \in \mathbb{N}$

$$\operatorname{Img}\mu_{\mathsf{reg}}^{\leq dL} \cap k[Y] = \ker\phi^{\leq L}. \tag{1.363}$$

Proof. Note that we have

$$\operatorname{Img}\mu_{\mathsf{reg}} \cap k[Y] = (f_1 - Y_1,\ldots,f_m - Y_m) \cap k[Y_1,\ldots,Y_m]. \tag{1.364}$$

Set $I := (f_1 - Y_1,\ldots,f_m - Y_m)$. For the first claim, it suffices to show that the elimination ideal $I \cap k[Y]$ equals the kernel of ϕ. Let $h \in \ker\phi \subset k[Y]$ and consider it as element of $k[X,Y]$. We have modulo I

$$h(Y_1,\ldots,Y_m) + I = h(f_1(X),\ldots,f_m(X)) + I = I. \tag{1.365}$$

Hence, h lies in $I \cap k[Y]$. To prove that $I \cap k[Y]$ is contained in $\ker\phi$, we will extend ϕ on $k[X,Y]$ by

$$\overline{\phi} : k[X,Y] \longrightarrow k[X] \tag{1.366}$$

$$X_i \longmapsto X_i \tag{1.367}$$

$$Y_j \longmapsto f_j(X). \tag{1.368}$$

I.e., $\overline{\phi}$ substitutes the variable Y_j by f_j, but does not change the variables $X_1,\ldots,X_n$. Hence, we have $\ker\overline{\phi} \cap k[Y] = \ker\phi$, and it suffices to prove that I is contained in $\ker\overline{\phi}$. This is indeed the fact, since $\overline{\phi}$ maps each generator $f_i - Y_i$ of I to zero.

To prove the second equality, we first note that each image element of $\mu_{\mathsf{reg}}^{\leq dL}$ can have degree at most dL. Since the Y variables have degree d in $k[X,Y]$, each element in $\operatorname{Img} \mu_{\mathsf{reg}}^{\leq dL} \cap k[Y]$ can be written as linear combination of products of at most L Y-variables and must, hence, lie in $\ker \phi^{\leq L}$. On the other hand, let $h \in \ker \phi^{\leq L}$. By an inductive argument, one can show for any $g \in k[X,Y]^{\leq dL} \cap k[Y]$ and $i \in [m]$ that

$$g(f_1, \ldots, f_{i-1}, Y_i, Y_{i+1}, \ldots, Y_m) - g(f_1, \ldots, f_{i-1}, f_i, Y_{i+1}, \ldots, Y_m) \in \operatorname{Img} \mu_{\mathsf{reg}}^{\leq dL}.$$

For $h \in \ker \phi^{\leq L}$, it follows

$$\begin{aligned} h(Y) &= h(Y_1, \ldots, Y_m) - h(f_1, \ldots, f_m) \\ =& \sum_{i=1}^{m} h(f_1, \ldots, f_{i-1}, Y_i, Y_{i+1}, \ldots, Y_m) - h(f_1, \ldots, f_{i-1}, f_i, Y_{i+1}, \ldots, Y_m). \end{aligned}$$

Since each difference lies in $\operatorname{Img} \mu_{\mathsf{reg}}^{\leq dL}$, h must lie in $\operatorname{Img} \mu_{\mathsf{reg}}^{\leq dL}$, too. □

We will further need the regular Macaulay map with respect to the field equations, if k is finite.

Definition 19. Let k be a field of finite size q, and let $f_1, \ldots, f_m \in k[X]^{\leq d}$. We define the **regular Macaulay map with field equations** for the polynomials $f_1, \ldots, f_m$ by

$$\begin{aligned} \mu_{q,\mathsf{reg}} : (k[X,Y])^{m+n} &\longrightarrow k[X,Y] \\ (g_1, \ldots, g_{m+n}) &\longrightarrow \sum_{i=1}^{m} g_i(X,Y) \cdot (f_i(X) - Y_i) + \sum_{i=1}^{n} g_{m+i}(X,Y) \cdot (X_i^q - X_i) \end{aligned}$$

Up to degree $L \in \mathbb{N}$, the regular Macaulay map with field equations is given by its restriction

$$\mu_{q,\mathsf{reg}}^{\leq L} : \left(k[X,Y]^{\leq L-d}\right)^m \times \left(k[X,Y]^{\leq L-q}\right)^n \longrightarrow k[X,Y]^{\leq L}, \tag{1.369}$$

where we set $k[X,Y]^{\leq a} := 0$ for $a < 0$.

Lemma 44. *Let k be a field of size q, and $f_1, \ldots, f_m \in k[X]^{\leq d}$. Let μ_{reg} be the regular Macaulay map for $f_1, \ldots, f_m$ with the field equations $X_1^q - X_1, \ldots, X_n^q - X_n$ of k. We have for each $L \in \mathbb{N}$*

$$\operatorname{Img} \mu_{q,\mathsf{reg}}^{\leq dL} \cap k[Y] = \left(\phi^{\leq L}\right)^{-1}(I_q) \tag{1.370}$$

where $I_q = (X_1^q - X_1, \ldots, X_m^q - X_m)$.

Proof. Let μ_{reg} be the regular Macaulay map for $f_1, \ldots, f_m$ *without* field equations, and let $\overline{\phi} : k[X,Y] \to k[X]$ be the extension of the dual morphism ϕ to $k[X,Y]$. We have shown in the proof of Lemma 43

$$\operatorname{Img} \mu_{\mathsf{reg}} = \ker \overline{\phi}. \tag{1.371}$$

Because of the claim of Lemma 43, this equality stays true when we restrict it to degree dL, i.e., we have

$$\operatorname{Img} \mu_{\mathsf{reg}}^{\leq dL} = \ker \overline{\phi}^{\leq L}. \tag{1.372}$$

To prove the statement of this lemma, it suffices to show

$$\operatorname{Img} \mu_{q,\mathsf{reg}}^{\leq dL} = \left(\overline{\phi}^{\leq L}\right)^{-1}(I_q) \tag{1.373}$$

for $I_q = (X_1^q - X_1, \ldots, X_n^q - X_n) \subset k[X]$. Set $I_X^{\leq dL} := (X_1^q - X_1, \ldots, X_n^q - X_n) \cap k[X,Y]^{\leq dL}$, then it is easy to see that we have

$$\operatorname{Img} \mu_{q,\mathsf{reg}}^{\leq dL} = \operatorname{Img} \mu_{\mathsf{reg}}^{\leq dL} + I_X^{\leq dL}. \tag{1.374}$$

On the other hand, we claim that

$$\left(\overline{\phi}^{\leq L}\right)^{-1}(I_q) = \ker \overline{\phi}^{\leq L} + I_X^{\leq dL}. \tag{1.375}$$

Since $\overline{\phi}$ maps $\ker \overline{\phi}^{\leq L}$ to zero and $I_X^{\leq dL}$ to itself, it follows that $\ker \overline{\phi}^{\leq L} + I_X^{\leq dL}$ is contained in $\left(\overline{\phi}^{\leq L}\right)^{-1}(I_q)$. Any element $h \in k[X,Y]^{\leq dL}$ with $\overline{\phi}(h) \in I_q$ can be written as

$$h = h - \overline{\phi}(h) + \overline{\phi}(h), \tag{1.376}$$

where $h - \overline{\phi}(h)$ must lie in $\ker \overline{\phi}^{\leq L}$, and $\overline{\phi}(h)$ must lie in $I_X^{\leq dL}$. Hence, Eq. (1.375) is true. Eq. (1.373) now follows, since we have

$$\begin{aligned} \operatorname{Img} \mu_{q,\mathsf{reg}}^{\leq dL} &\overset{Eq.~(1.374)}{=} \operatorname{Img} \mu_{\mathsf{reg}}^{\leq dL} + I_X^{\leq dL} && (1.377)\\ &\overset{Eq.~(1.372)}{=} \left(\overline{\phi}^{\leq L}\right)^{-1}(I_q) + I_X^{\leq dL} && (1.378)\\ &\overset{Eq.~(1.375)}{=} \left(\overline{\phi}^{\leq L}\right)^{-1}(I_q). && \square \end{aligned}$$

Finally, we observe the following simple Weil restriction:

Lemma 45. *Let $k \subset \overline{k}$ be a field extension of degree r with generator ζ. For every polynomial $h \in \overline{k}[Y_1', \ldots, Y_{m'}']$ of degree L, there are r polynomials $u_1, \ldots, u_r \in k[Y_1, \ldots, Y_{rm'}]$ of degree L s.t.*

$$h(y_1', \ldots, y_{m'}') = u_1(Y) + \zeta \cdot u_2(Y) + \ldots + \zeta^{r-1} u_r(Y) \tag{1.379}$$

where we set for $j \in [m']$

$$y_j(Y)' := Y_{r\cdot(j-1)+1} + \zeta \cdot Y_{r\cdot(j-1)+2} + \ldots + \zeta^{r-1} \cdot Y_{r\cdot j}. \tag{1.380}$$

By using the isomorphism $\psi : k^r \to \overline{k}$ of Proposition 33, we can rewrite Eq. (1.379) as

$$\begin{aligned} h(\psi(Y)) =& h\big(\psi(Y_1, \ldots, Y_r), \ldots, \psi(Y_{r\cdot(m'-1)+1}, \ldots, Y_{r\cdot m'})\big) && (1.381)\\ =& \psi(u_1(Y), \ldots, u_r(Y)) = \psi(u_1, \ldots, u_r)(Y) = \psi(u)(Y). && (1.382) \end{aligned}$$

Proof. Set $m = rm'$. We want to prove the existence of a map $u : k^m \to k^r$ of degree L s.t. the abbreviated equation

$$h(\psi(Y)) = \psi(u)(Y) \tag{1.383}$$

holds. Note that $\psi(Y)$ is a collection of m' polynomials of degree 1 in $\overline{k}[Y_1, \ldots, Y_m]$. Hence, $h(\psi(Y))$ is a polynomial of degree L in $\overline{k}[Y_1, \ldots, Y_m]$. For each multi-index $\alpha \in \mathbb{N}_0^m$, let $c_\alpha \in \overline{k}$ be the corresponding coefficient of $h(\psi(Y))$ s.t. we have

$$h(\psi(Y)) = \sum_\alpha c_\alpha Y^\alpha. \tag{1.384}$$

For each α, set $(c_{\alpha,1}, \ldots, c_{\alpha,r}) := \psi^{-1}(c_\alpha) \in k^r$. I.e., we have

$$c_{\alpha,1} + \zeta \cdot c_{\alpha,2} + \ldots + \zeta^{r-1} \cdot c_{\alpha,r} = \psi(c_{\alpha,1}, \ldots, c_{\alpha,r}) = c_\alpha. \tag{1.385}$$

Finally, for $i \in [r]$, set

$$u_i(Y_1, \ldots, Y_m) := \sum_\alpha c_{\alpha,i} Y^\alpha. \tag{1.386}$$

Because of the $k[X]$-linearity of ψ, it then follows

$$\begin{aligned} h(\psi(Y)) =& \sum_\alpha c_\alpha Y^\alpha = \sum_\alpha \psi(c_{\alpha,1}, \ldots, c_{\alpha,r}) \cdot Y^\alpha & (1.387) \\ =& \psi\left(\sum_\alpha c_{\alpha,1} Y^\alpha, \ldots, \sum_\alpha c_{\alpha,r} Y^\alpha\right) = \psi(u_1, \ldots, u_r)(Y). & \square \end{aligned}$$

Finally, to prove Theorem 41, we need the following observations:

Proposition 46. *Let $f_1, \ldots, f_m \in k[X]$ be of degree d and let $y \in k^m$. Let μ_{reg} be the regular Macaulay map for $f_1, \ldots, f_m$ and let μ be the (non-regular) Macaulay map for $f_1, \ldots, f_m$ and y. Let*

$$E_y : k[X, Y] \longrightarrow k[X] \tag{1.388}$$

$$X_i \longmapsto X_i \tag{1.389}$$

$$Y_j \longmapsto y_j \tag{1.390}$$

be the evaluation morphism at y.

We have

$$E_y \circ \mu_{\mathsf{reg}} = \mu \circ E_y, \tag{1.391}$$

and for each $L \in \mathbb{N}$

$$E_y\big(\operatorname{Img} \mu_{\mathsf{reg}}^{\leq L}\big) = \operatorname{Img} \mu^{\leq L}. \tag{1.392}$$

Proof Theorem 41. Let $\mathcal{F} : k^n \to k^m$ be a PRG of degree d over a field q. Draw $F \leftarrow \mathcal{F}$ and $y \leftarrow k^m$. Note that $\mathcal{M}$ from Algorithm 7 is a refutation algorithm, i.e., it will never reject an image of F. Hence, it suffices to prove lower bounds for the probability that $\mathcal{M}$ outputs 1 on input (F, y). Finally, note that $\mathcal{M}$ outputs 1 if 1 lies in the image of $\mu^{\leq dD}$.

We prove each point separately:

1. Let $\phi : k[Y] \to k[X]$ be the dual morphism of F and set

$$L := \left\lceil 2^{\frac{d}{d-1}} \cdot \left(\frac{n^d}{m(n)}\right)^{1/(d-1)} \right\rceil. \tag{1.393}$$

According to Theorem 10, ϕ has a non-zero kernel element h of degree $\leq L$. Because of Lemma 43, h lies in $\operatorname{Img}\mu_{\mathsf{reg}}^{\leq dL}$. Hence, $h(y)$ lies in $\operatorname{Img}\mu^{\leq dL} \subset \operatorname{Img}\mu^{\leq dD}$. According to the Schwartz-Zippel Lemma 2, $h(y)$ lies in $k^{\times}$ with probability at least $\geq 1 - L/\#k$. Hence, with probability $\geq 1 - L/\#k$, the image of $\mu^{\leq dD}$ contains a non-zero scalar and $\mathcal{M}$ outputs 1.

2. For simplicity, assume that r divides m and set $m' := m/r$. Let $\overline{k}$ be an extension field of k of degree r and consider the map $\psi \circ F : k^n \to \overline{k}^{m'}$. Let $\phi : \overline{k}[Y_1', \ldots, Y_{m'}'] \to \overline{k}[X]$ be its dual morphism. Because of Theorem 10, there is a non-trivial kernel element $h \in \ker \phi$ of degree

$$\deg h \leq \left\lceil 2^{\frac{d}{d-1}} \cdot \left(\frac{r(n) \cdot n^d}{m(n) - r(n)} \right)^{\frac{1}{d-1}} \right\rceil =: L. \tag{1.394}$$

Because of Lemma 45, we can make a Weil descent and deduce the existence of polynomials $u_1, \ldots, u_r \in k[Y_1, \ldots, Y_m]$ of degree $\leq D$ s.t.

$$h \circ \psi = \psi(u_1, \ldots, u_r). \tag{1.395}$$

Note that we have for $u = (u_1, \ldots, u_r)$

$$0 = \phi(h) = h(\psi(F(X))) = \psi(u(F(X))). \tag{1.396}$$

Since ψ is a bijection, it follows $u_i(f_1, \ldots, f_m) = 0$ for each $i \in [r]$. Hence, each of the polynomials $u_1, \ldots, u_r$ is an algebraic relation of F of degree $\leq D$ and must lie in the image of $\mu_{\mathsf{reg}}^{\leq dD}$. In particular, $\operatorname{Img}\mu^{\leq dD}$ contains the values $u_1(y), \ldots, u_r(y)$.

Since y is uniformly distributed in k^m, $\psi(y)$ is uniformly distributed in $\overline{k}^{m'}$. The Schwartz-Zippel Lemma 2 implies that $h(\psi(y))$ is non-zero with probability $\geq 1 - \deg h/\#\overline{k}$. Since $h(\psi(y)) = \psi(u_1(y), \ldots, u_r(y))$, it follows that one u_i is not zero at y with probability $\geq 1 - \deg h/\#\overline{k}$. Since $u_i(y) \neq 0$ with probability $\geq 1 - \deg h/\#\overline{k}$, the (non-regular) Macaulay image $\mu^{\leq D}$ must contain a non-zero value $u_i(y)$ with probability $\geq 1 - \deg h/\#\overline{k}$.

3. Again, let $\phi : k[Y] \to k[X]$ be the dual morphism of F and set

$$L := \left\lceil 2^{\frac{d+1}{d-1}} \cdot \left(\frac{n^d}{m(n)} \right)^{1/(d-1)} \right\rceil. \tag{1.397}$$

Theorem 10 implies that $\ker \phi^{-1}(I_q)$ has an element h of degree $\leq L$ s.t. $h \notin I_q$. Because of Lemma 44, h lies in the image of $\mu_{q,\mathsf{reg}}^{\leq dL}$. Hence, $h(y)$ lies in the image of $\mu^{\leq dL}$. Lemma 5 implies that h does not vanish with probability $\geq q^{-\deg h}$. Hence, the advantage of $\mathcal{M}$ is lower bounded by $\geq q^{-\deg h}$. □

To prove Theorem 42, we need the following small lemma:

Lemma 47. *Let k be a field of size q and let $x_1 \in k$. There are scalars $\alpha_z \in k$, for $z \in k \setminus \{x_1\}$, s.t. we have*

$$X_1 - x_1 = \sum_{z \in k \setminus \{x_1\}} \alpha_z \cdot \frac{X_1^q - X_1}{X_1 - z}. \tag{1.398}$$

Proof. Without loss of generality, we can assume that $x_1 = 0$. Note that $\frac{X_1^q - X_1}{X_1 - z} = \prod_{y \in k \setminus \{z\}} (X_1 - y)$ and set for $z \neq 0$

$$\alpha_z := \prod_{y \in k \setminus \{0,z\}} \frac{1}{z-y}. \tag{1.399}$$

We then have for each $u \in k \setminus \{0\}$

$$\alpha_z \cdot \frac{u^{q-1}-1}{u-z} = \begin{cases} 0, & \text{if } u \in k \setminus \{0,z\}, \\ 1, & \text{if } u = z. \end{cases} \tag{1.400}$$

This is, because $\frac{u^{q-1}-1}{u-z} = \prod_{v \in k \setminus \{0,z\}} (u-v)$. It follows that the polynomial

$$g(X_1) = \sum_{z \in k \setminus \{x_1\}} \alpha_z \cdot \frac{X_1^{q-1}-1}{X_1 - z} \tag{1.401}$$

is 1 on $k \setminus \{0\}$. Since the degree of g is $q-2$, it follows that g must be the constant 1. Hence, we have

$$\sum_{z \in k \setminus \{x_1\}} \alpha_z \cdot \frac{X_1^q - X_1}{X_1 - z} = X_1 \cdot g(X_1) = X_1. \qquad \square$$

Proof Theorem 42. Let $q = q(n)$ be the size of k. Assume, for simplicity, again that $r(n)$ always divides $m(n)$ and set $m' := m(n)/r(n)$ for a given $n \in \mathbb{N}$. Let $x \in k^n$ and draw $F \leftarrow \mathcal{F}$, which is of type $k^n \to k^m$ and of degree d. Set $y := F(x)$. Note that $\mathcal{M}^{\mathsf{solv}}$ computes the image of the Macaulay map μ for the polynomial equation system

$$F(X) = y, \tag{1.402}$$
$$X^q - X = 0. \tag{1.403}$$

Let $\phi : k[Y] \to k[X]$ be the dual morphism of F.

Let us consider the first coordinate X_1 of F. Because of our reasoning in the proof of Lemma 40 and the second part of Theorem 41, we know that, for each $z \neq x_1$, the ideal generated by

$$f_1(z, X_2, \ldots, X_n) - y_1, \ldots, f_m(z, X_2, \ldots, X_n) - y_1 \tag{1.404}$$

must yield 1 with probability $\geq 1 - o\left(\frac{1}{qn}\right)$ at degree $D(n)$. In particular, there are polynomials $g_1, \ldots, g_m \in k[X]^{\leq dD-d}$ and $g_0 \in k[X]^{\leq dD-1}$ s.t.

$$1 = g_0(X) \cdot (X_1 - z) + g_1(X) \cdot (f_1(X) - y_1) + \ldots + g_m(X) \cdot (f_m(X) - y_m).$$

By multiplying both sides with $\prod_{s \in k \setminus \{z\}} (X_1 - s)$, it follows that

$$\prod_{s \in k \setminus \{z\}} (X_1 - s) = g_0(X) \cdot (X_1^q - X_1)$$
$$+ \prod_{s \in k \setminus \{z\}} (X_1 - s) \cdot (g_1(X) \cdot (f_1(X) - y_1) + \ldots + g_m(X) \cdot (f_m(X) - y_m))$$

must lie in $\operatorname{Img}\mu^{dD+q-1}$. Hence, with probability $\geq 1 - o\big(\frac{1}{n}\big)$, $\operatorname{Img}\mu^{dD+q-1}$ contains

$$\prod_{s \in k \setminus \{z\}} (X_1 - s) = \frac{X_1^q - X_1}{X_1 - z} \tag{1.405}$$

for every $z \in k \setminus \{x_1\}$. Lemma 47 implies that, with probability $\geq 1 - o\big(\frac{1}{n}\big)$, $\operatorname{Img}\mu^{dD+q-1}$ must contain the equation $X_1 - x_1$. Analogously, for each $i \in [n]$, $\operatorname{Img}\mu^{dD+q-1}$ contains $X_i - x_i$, with probability $\geq 1 - o\big(\frac{1}{n}\big)$. Hence, $\mathcal{M}^{\mathsf{solv}}$ will yield the correct solution $x \in k^n$ with probability $\geq 1 - o(1)$. □

1.5.2 Which Algorithm is Faster?

We will compare here the time complexity of the algebraic relation-based refutation Algorithm 4 $\mathcal{A}$, with the time complexity of the Macaulay matrix-based refutation Algorithm 7 $\mathcal{M}$. Comparisons for the Algorithms 5, 6 and 8 follow analogously. We will only compare the upper bounds that we could prove in Theorems 29 and 41, and we will ignore the fact that $\mathcal{M}$ performs multiple iterations and, instead, assume that it directly computes the Macaulay matrix M_{dD} for a sufficiently large D. Further, we will ignore all bit operations performed by both algorithms and just consider the number of arithmetic operations over the corresponding field.

Let $\mathcal{F}$ be a PRG of sufficient stretch $m \in \omega(n)$ and degree d over a field k of sufficient size $\#k \geq n$. Let $D = \Big\lceil 2^{\frac{d}{d-1}} \cdot (n^d/m)^{1/(d-1)} \Big\rceil$. (However, note that any D s.t. $\phi^{\leq D}$ is not injective would suffice to guarantee a high advantage of the algorithms $\mathcal{A}$ and $\mathcal{M}$ in Game 2 against $\mathcal{F}$.)

The offline time complexity of $\mathcal{A}$ is dominated by applying Gaussian elimination on a dense matrix of shape $\binom{m+D}{D} \times \binom{n+dD}{dD}$. Since D is chosen s.t. $\binom{m+D}{D} > \binom{n+dD}{dD}$, $\mathcal{A}$'s offline algorithm performs $O\Big(\binom{m+D}{D} \cdot \binom{n+dD}{dD}^2\Big)$ arithmetic operations over k. $\mathcal{A}$'s online algorithm only needs to evaluate a degree-D polynomial on a vector of length m, which can be performed by $O\Big(D \cdot \binom{m+D}{D}\Big)$ arithmetic operations over k.

On the other hand, $\mathcal{M}$ does not have an offline part. Its online part is dominated by applying Gaussian elimination on the Macaulay matrix M_{dD}, which is of shape $m \cdot \binom{n+dD-d}{dD-d} \times \binom{n+dD}{dD}$. However, M_{dD} is sparse, each row of M_{dD} contains at most $\binom{n+d}{d}$ non-zero entries. In this case, one usually applies the block Wiedemann algorithm, given by Coppersmith [Cop94], which is a generalization of the algorithm of Wiedemann [Wie86]. The block Wiedemann algorithm is randomized and searches for a non-trivial kernel vector of a square $N \times N$-matrix that contains d entries in each column by performing $O(N^2 d)$ arithmetic operations. For pathological matrices, it may not always succeed, however, in general, its success probability is lower bounded [Kal95; Vil97; HJS16; HJS22]. Now, note that the Macaulay matrix M_{dD} is not square. Further, we need to determine if a unit vector b, which corresponds to the constant polynomial 1, lies in the row span of M_{dD}. The typical approach [CCNY12; Beu21; Beu22] in this case seems to be to just take a submatrix B of M_{dD} of shape $\Big(\binom{n+dD}{dD} - 1\Big) \times \Big(\binom{n+dD}{dD} - 1\Big)$ that contains all columns of M_{dD} corresponding to non-constant monomials and a random subset of rows. Then, one uses the algorithm of Coppersmith [Cop94] to find a non-trivial kernel-vector of B^T with high probability. Finally, one checks if this vector corresponds to a non-zero constant when taking the absolute coefficients of the corresponding polynomials into account. Since B^T has at most $\binom{n+d}{d} - 1$ entries in each column, the time complexity of this procedure is given by

$$O\left(\left(\binom{n+dD}{dD} - 1\right) \cdot \left(\binom{n+dD}{dD} - 1\right) \cdot \left(\binom{n+d}{d} - 1\right)\right) \tag{1.406}$$

$$=O\left(\binom{n+d}{d} \cdot \binom{n+dD}{dD}^2\right). \tag{1.407}$$

We assume that $\binom{n+dD}{dD}$ is close to $\binom{m+D}{D}$, since D is chosen minimal with $\binom{m+D}{D} > \binom{n+dD}{dD}$. With this approximation, we can see in Table 1.1, that the total time complexity of $\mathcal{M}$ is, presumably, by a factor of $2/3$ in the exponent smaller than the total time complexity of $\mathcal{A}$. However, the online time complexity of $\mathcal{A}$ is by an amortized factor of $1/2$ in the exponent smaller than the online time complexity of $\mathcal{M}$.

To the question, which algorithm is faster, we can give the following answer: if there is no preprocessing phase, then the Macaulay matrix-based refutation Algorithm 7 is faster than the Algorithms 4 and 5 based on algebraic relations. However, when we consider fixed PRGs where, for each $n \in \mathbb{N}$, always the same fixed deterministic function $F_n : k^n \to k^{m(n)}$ is used, then Algorithms 4 and 5 give us a non-uniform adversary in Game 2 that, presumably, will outperform its Macaulay matrix-based counterpart.

In the case of solving an equation system $F(X) = y$, our algebraic relation-based Algorithm 6 does not offer a preprocessing phase, since its advantage is based on the randomness of F. Hence, for these problems, the Macaulay matrix-based solving Algorithm 8 is always preferable.

Operations over k	Algorithm 4: $\mathcal{A}$	Algorithm 7: $\mathcal{M}$
Offline	$O\left(\binom{m+D}{D} \cdot \binom{n+dD}{dD}^2\right)$	0
Online	$O\left(D \cdot \binom{m+D}{D}\right)$	$O\left(\binom{n+d}{d} \cdot \binom{n+dD}{dD}^2\right)$

Table 1.1: Upper bounds on the time complexity of the PRG attacker of Algorithm 4 and the Macaulay matrix-based refutation Algorithm 7 in Game 2 against a PRG $\mathcal{F} : k^n \to k^m$ of degree d and stretch $m \in \omega(n)$.

On Lower Bounds. To give a complete comparison between the algorithms in Sections 1.3 and 1.4 and the Macaulay matrix-based algorithms here, it would be necessary to know lower bounds for the time complexity of both kinds of algorithms. As discussed in Section 1.2.2, for the algorithms based on algebraic relations, one would need to determine the minimal value D for $n, m, d \in \mathbb{N}$ s.t. the dual morphism of a uniformly random polynomial map $F : k^n \to k^m$ of degree d over an exponentially large field k has a non-trivial kernel element of degree $\leq D$ with noticeable probability.

Now, given a polynomial map $F : k^n \to k^m$ with dual morphism ϕ, a random target point $y \leftarrow k^m$ and a minimal $D \in \mathbb{N}$ s.t. $\ker \phi^{\leq D} \neq 0$, we know that the

Macaulay map $\mu^{\leq dL}$ up to degree dL must contain 1 in its image with high probability, if k is large enough. However, it may be that the image of the Macaulay map $\mu^{\leq i}$ up to degree i for some $i < dL$ already contains 1 and that Algorithm 7 terminates before it performs dL iterations. One can ask how likely this is going to occur if we draw $y \leftarrow k^m$ uniformly at random over an exponentially large field k. With regard to Lemma 43 that claims for each $L \in \mathbb{N}$

$$\operatorname{Img} \mu_{\mathsf{reg}}^{\leq dL} \cap k[Y] = \ker \phi^{\leq L}, \tag{1.408}$$

we can ask the following question:

Question 5. *Let $F : k^n \to k^m$ be polynomial of degree d over a field k of size $q \geq 2^n$. Let μ_{reg} be the regular Macaulay map for F. Let $L \in \mathbb{N}$ be maximal with*

$$\operatorname{Img} \mu_{\mathsf{reg}}^{\leq dL} \cap k[Y] = 0. \tag{1.409}$$

Draw $y \leftarrow k^m$ uniformly at random and let $E_y : k[X, Y] \to k[X]$ be the evaluation morphism that maps Y_i to y_i. Is the probability that we have

$$1 \in \operatorname{Img} \mu^{\leq dL} = E_y(\operatorname{Img} \mu_{\mathsf{reg}}^{\leq dL}) \tag{1.410}$$

noticeable?

If the answer to the above question is positive, then, in a non-negligible number of cases, the Macaulay map μ can find contradictions of the equation system $F(X) = y$ *faster* than a non-trivial kernel element of the dual morphism ϕ of F.

Regarding the prototype Macaulay matrix-based Algorithms 7 and 8 we gave in this section, note that they do not incorporate techniques used by XL [CKPS00; YC05] and Mutant-XL [DBMMW08; MMDB08] such as extracting roots of univariate polynomials and extending the Macaulay matrix in the event of degree falls. In particular, in the case of the multivariate search problem of Definition 15, it would be desirable to know if those techniques can substantially accelerate the growth of the rank of the Macaulay matrix. Hence, we ask here how likely it is that those techniques are applicable over exponentially large fields:

Question 6. *Let k be a finite field of size $q \geq 2^n$. Let $F : k^n \to k^m$ be a uniformly random map of degree d and let $x \leftarrow k^n$. Let L s.t.*

$$\binom{m+L}{L} < \binom{n+dL}{dL}. \tag{1.411}$$

Is the probability that the image of $\mu^{\leq dL}$ contains a non-zero univariate polynomial noticeable?

Is the probability that a degree fall during the computation of the spaces

$$\operatorname{Img} \mu^{\leq d}, \operatorname{Img} \mu^{\leq d+1}, \ldots \operatorname{Img} \mu^{\leq dL} \tag{1.412}$$

occurs noticeable? I.e., is the probability that there exists an $i \in \{d, \ldots, dL-1\}$ s.t.

$$\operatorname{Img} \mu^{\leq i} \subsetneq \operatorname{Img} \mu^{\leq i+1} \cap k[X]^{\leq i} \tag{1.413}$$

noticeable?

Finally, all questions we asked in this chapter dealt with lower bounds of our algorithms in the case of exponentially large fields. We could also ask for lower bounds for the PRG adversaries of Section 1.3 and of this section in the case where k is small. Note that in the case of small fields one cannot give bounds by considering the case of generic polynomials. An important lower bound for the field $k = \mathbb{Z}_2$ has been given by Applebaum and Lovett [AL16], who proved that each algebraic attack on a random local function of sufficient rational degree must be subexponential:

Theorem 48 (Applebaum and Lovett [AL16] Theorem 5.5)**.** *Let $m = \lceil n^{1+e} \rceil$ for constant $e > 0$. Let $d > 9 + 8e$ be constant. There is a PRG $\mathcal{F} : \{0,1\}^n \to \{0,1\}^m$ of rational degree d and a function $D \in \Omega(n^{1-16e/(d-1)})$ s.t. we have with high probability over $F \leftarrow \mathcal{F}$ and $y \leftarrow \{0,1\}^m$*

$$1 \notin \operatorname{Img} \mu^{\leq dD}, \tag{1.414}$$

where μ is the Macaulay map for the equation system $F(X) = y$.

Chapter 2

Functional Encryption

In this chapter, we will derive lower bounds for lattice-based functional encryption schemes. We will follow a bottom-up approach and start with the elementary study of the multivariate Vandermonde matrix in Section 2.2. In Section 2.3, we will continue by studying the task of distinguishing two different distributions by approximating mean squares. In Section 2.4, we will apply Vandermonde matrices and mean square distinguishing to attack secret-key encryption schemes of simple offline/online design. Finally, in Section 2.5, we will be able to derive lower bounds for FE.

Let us start by reviewing definitions and lemmas that are in use over this chapter.

2.1 Preliminaries

2.1.1 Mathematical Preliminaries

Absolute Value of Residue Classes

Definition 20. Let $q \in \mathbb{N}$. Given an element $x \in \mathbb{Z}_q$, we define its **absolute value modulo** q by

$$|x \bmod q| := \min_{y \in x + q\mathbb{Z}} |y| \in \left\{0, \ldots, \left\lfloor \frac{q}{2} \right\rfloor \right\}. \tag{2.1}$$

For an element $v = (v_1, \ldots, v_m) \in \mathbb{Z}_q^m$, we define its **infinity norm modulo** q by

$$||v \bmod q||_\infty := \max_{i \in [m]} |v_i \bmod q|. \tag{2.2}$$

Absolute values modulo q behave very similar to the normal absolute value over the reals. It is easy to show that they are positive-definite and fulfil the triangle inequality. While they are not fully homogeneous, they fulfil a relaxed version of homogeneity that is given in Eq. (2.5).

Proposition 49. *For $x, y \in \mathbb{Z}_q$ and $c \in \mathbb{Z}$ we have*

$$|x \bmod q| = 0 \iff x \in q\mathbb{Z}, \tag{2.3}$$

$$|(x + y) \bmod q| \leq |x \bmod q| + |y \bmod q|, \tag{2.4}$$

$$|(c \cdot x) \bmod q| \leq |c| \cdot |x \bmod q|. \tag{2.5}$$

In this chapter, we will associate the valuation $|\_ \bmod q|$ with the ring $\mathbb{Z}_q$ and simply write $|x|$ instead of $|x \bmod q|$ when we are given an element $x \in \mathbb{Z}_q$. We will also simply write $||v||_\infty$ instead of $||v \bmod q||_\infty$ when $v \in \mathbb{Z}_q^m$.

2.1.2 Stochastic Preliminaries

We will show here two technical inequalities that involve discrete distributions:

Lemma 50. *Let $n \in \mathbb{N}$ and let $\mathcal{A}$ be a discrete memoryless distribution with support in some set A. Let $(\mathcal{B}_a)_{a\in A}$ and $(\mathcal{C}_a)_{a\in A}$ be families of discrete memoryless distributions with support in some set B. Sample $a_1, a_2 \leftarrow \mathcal{A}$, $b \leftarrow \mathcal{B}_{a_1}$ and $c \leftarrow \mathcal{C}_{a_2}$ and set*

$$\varepsilon := \Delta((a_1, b), (a_2, c)). \tag{2.6}$$

For $a_1, a_2 \leftarrow \mathcal{A}$, $b_1, \dots, b_n \leftarrow \mathcal{B}_{a_1}$ and $c_1, \dots, c_n \leftarrow \mathcal{C}_{a_2}$, we have

$$\Delta((a_1, b_1, \dots, b_n), (a_2, c_1, \dots, c_n)) \leq n \cdot \varepsilon. \tag{2.7}$$

Proof. For $a \in A$, denote by $p(a)$ the probability that a gets sampled by $\mathcal{A}$. Further, for $b \in B$, denote by $q(b|a)$ the probability that b gets sampled by $\mathcal{B}_a$, and denote by $r(b|a)$ the probability that b gets sampled by $\mathcal{C}_a$.

For $a_1, a_2 \leftarrow \mathcal{A}$, $b \leftarrow \mathcal{B}_{a_1}$ and $c \leftarrow \mathcal{B}_{a_2}$, we have

$$\begin{aligned}
\varepsilon =& \Delta((a_1, b), (a_2, c)) && (2.8)\\
=& \frac{1}{2} \sum_{x\in A, y\in B} |p(x) \cdot q(y|x) - p(x) \cdot r(y|x)| && (2.9)\\
=& \sum_{x\in A} p(x) \cdot \left(\frac{1}{2} \sum_{y\in B} |q(y|x) - r(y|x)| \right) && (2.10)\\
=& \sum_{x\in A} p(x) \cdot \Delta(\mathcal{B}_x, \mathcal{C}_x). && (2.11)
\end{aligned}$$

Similarly, for $b_1, \dots, b_n \leftarrow \mathcal{B}_{a_1}$ and $c_1, \dots, c_n \leftarrow \mathcal{C}_{a_2}$, we get

$$\begin{aligned}
&\Delta((a_1, b_1, \dots, b_n), (a_2, c_1, \dots, c_n)) && (2.12)\\
=& \frac{1}{2} \sum_{x\in A, y_1,\dots,y_n\in B} |p(x) \cdot q(y_1|x) \cdots q(y_n|x) - p(x) \cdot r(y_1|x) \cdots r(y_n|x)| && (2.13)\\
=& \sum_{x\in A} p(x) \cdot \left(\frac{1}{2} \sum_{y\in B} |q(y_1|x) \cdots q(y_n|x) - r(y_1|x) \cdots r(y_n|x)| \right) && (2.14)\\
=& \sum_{x\in A} p(x) \cdot \Delta(\mathcal{B}_x^n, \mathcal{C}_x^n), && (2.15)
\end{aligned}$$

where $\mathcal{B}_x^n$ and $\mathcal{C}_x^n$ denote the n-fold product distribution of $\mathcal{B}_x$ and $\mathcal{C}_x$, respectively. Since the statistical distance adheres to the triangle inequality, we can finally deduce

$$\begin{aligned}
&\Delta((a_1, b_1, \dots, b_n), (a_2, c_1, \dots, c_n)) && (2.16)\\
=&\sum_{x\in A} p(x) \cdot \Delta(\mathcal{B}_x^n, \mathcal{C}_x^n) && (2.17)\\
\leq&\sum_{x\in A} p(x) \cdot n \cdot \Delta(\mathcal{B}_x, \mathcal{C}_x) && (2.18)\\
=&n \cdot \sum_{x\in A} p(x) \cdot \Delta(\mathcal{B}_x, \mathcal{C}_x) = n \cdot \varepsilon. && \square
\end{aligned}$$

Lemma 51. *Let A, B be some sets and let $\phi : A \times B \to \{\mathsf{TRUE}, \mathsf{FALSE}\}$ be a predicate resp. event. Let $\mathcal{A}$ be a discrete memoryless distribution over A, and let $(\mathcal{B}_a)_{a\in A}$ be a family of discrete memoryless distributions over B. Set $\rho := \Pr_{a\leftarrow\mathcal{A}, b\leftarrow\mathcal{B}_a}[\phi(a, b)]$. Then, we have for each $\varepsilon \in [0, \rho)$*

$$\Pr_{a\leftarrow\mathcal{A}}\left[\Pr_{b\leftarrow\mathcal{B}_a}[\phi(a, b)] \geq \varepsilon\right] > \frac{\rho - \varepsilon}{1 - \varepsilon}. \tag{2.19}$$

With $\varepsilon = \rho/2$, we get

$$\Pr_{a\leftarrow\mathcal{A}}\left[\Pr_{b\leftarrow\mathcal{B}_a}[\phi(a, b)] \geq \frac{\rho}{2}\right] \geq \frac{\rho}{2 - \rho} \geq \frac{\rho}{2}. \tag{2.20}$$

Proof. Denote by $p(a)$ the probability that a gets sampled by $\mathcal{A}$ and set $\rho_a := \Pr_{b\leftarrow\mathcal{B}_a}[\phi(a, b)]$. We have

$$\begin{aligned}
\rho =& \Pr_{a\leftarrow\mathcal{A}, b\leftarrow\mathcal{B}_a}[\phi(a, b)] && (2.21)\\
=&\sum_{a\in A} p(a) \cdot \Pr_{b\leftarrow\mathcal{B}_a}[\phi(a, b)] && (2.22)\\
=&\sum_{a\in A} p(a) \cdot \rho_a && (2.23)\\
=&\sum_{a\in A, \rho_a<\varepsilon} p(a) \cdot \rho_a + \sum_{a\in A, \rho_a\geq\varepsilon} p(a) \cdot \rho_a && (2.24)\\
<&\sum_{a\in A, \rho_a<\varepsilon} p(a) \cdot \varepsilon + \sum_{a\in A, \rho_a\geq\varepsilon} p(a) && (2.25)\\
=&\varepsilon \cdot \Pr_{a\leftarrow\mathcal{A}}[\rho_a < \varepsilon] + \Pr_{a\leftarrow\mathcal{A}}[\rho_a \geq \varepsilon] && (2.26)\\
=&\varepsilon \cdot \Big(1 - \Pr_{a\leftarrow\mathcal{A}}[\rho_a \geq \varepsilon]\Big) + \Pr_{a\leftarrow\mathcal{A}}[\rho_a \geq \varepsilon] && (2.27)\\
=&\varepsilon + (1 - \varepsilon) \cdot \Pr_{a\leftarrow\mathcal{A}}[\rho_a \geq \varepsilon], && (2.28)
\end{aligned}$$

which is equivalent to the inequality $\frac{\rho-\varepsilon}{1-\varepsilon} < \Pr_{a\leftarrow\mathcal{A}}[\rho_a \geq \varepsilon]$. □

2.1.3 Functional Encryption Schemes

Before we can formally define functional encryption schemes we first need to lay down the notion of *function spaces*. A function space formally specifies the functionalities that can be evaluated by an FE scheme.

Definition 21 (Message, Function and Value Spaces)**.** Let $\mathcal{X} = (\mathcal{X}_\lambda)_\lambda, \mathcal{Y} = (\mathcal{Y}_\lambda)_\lambda$ and $\mathcal{F} = (\mathcal{F}_\lambda)_\lambda$ be three families of sets that are parametrized by $\lambda \in \mathbb{N}$. Assume that there are $s_1, s_2, s_3 \in \mathsf{poly}(\lambda)$ s.t. we have for all $\lambda \in \mathbb{N}$

$$\mathcal{X}_\lambda \subseteq \{0,1\}^{s_1(\lambda)}, \qquad \mathcal{Y}_\lambda \subseteq \{0,1\}^{s_2(\lambda)}, \qquad \mathcal{F}_\lambda \subseteq \{0,1\}^{s_3(\lambda)}. \tag{2.29}$$

If, for each $\lambda \in \mathbb{N}$, each $f \in \mathcal{F}_\lambda$ describes a deterministic and efficient function of type $\mathcal{X}_\lambda \to \mathcal{Y}_\lambda$, then we call $\mathcal{F}$ a **function space**, $\mathcal{X}$ a **message space** and $\mathcal{Y}$ a **value space**. We will write in this case

$$\mathcal{F} : \mathcal{X} \longrightarrow \mathcal{Y}. \tag{2.30}$$

Note, that in the above definition we required that each $f \in \mathcal{F}_\lambda$ *describes* a function. Since an element $f \in \mathcal{F}_\lambda$ is a bit string, it of course cannot be a function in the mathematical sense. Formally, we require that for the function space $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$ there is a fixed algorithm Eval that is deterministic and efficient s.t. for each $\lambda \in \mathbb{N}$ and each pair $f \in \mathcal{F}_\lambda, x \in \mathcal{X}_\lambda$ the algorithm Eval on input f and x outputs an element in $\mathcal{Y}_\lambda$. In this case, the bit string f describes the function $\mathsf{Eval}(f, \_) : \mathcal{X}_\lambda \to \mathcal{Y}_\lambda$. In this chapter, we will informally identify each function description $f \in \mathcal{F}_\lambda$ with the mathematical function $\mathsf{Eval}(f, \_)$ and set $f(x) := \mathsf{Eval}(f, x)$ for $x \in \mathcal{X}_\lambda$.

We will also need to reason about subspaces of message spaces:

Definition 22 (Subspaces)**.** A **subspace** $\widetilde{\mathcal{X}} = (\widetilde{\mathcal{X}}_\lambda)_\lambda$ of a message space $\mathcal{X} = (\mathcal{X}_\lambda)_\lambda$ is a family of sets s.t. we have $\widetilde{\mathcal{X}}_\lambda \subseteq \mathcal{X}_\lambda$ for each $\lambda \in \mathbb{N}$. We will write in this case $\widetilde{\mathcal{X}} \subseteq \mathcal{X}$.

Let us now introduce functional encryption schemes. Note, that all FE schemes in this work are *symmetric*, i.e., there is no public key and only the master secret key-holder can encrypt messages.

Definition 23 (Functional Encryption)**.** A **functional encryption (FE)** scheme for a function space $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$ is a tuple of four stateless algorithms $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$. We will informally describe the four algorithms as follows:

Setup: On input the unary encoded security parameter 1^λ, Setup computes and outputs a master secret key msk.

KeyGen: On input a master secret key and a function $f \in \mathcal{F}_\lambda$, KeyGen computes and outputs a secret key sk_f for f.

Enc: On input a master secret key msk and a message $x \in \mathcal{X}_\lambda$, Enc computes and outputs a ciphertext ct_x for x.

Dec: On input a secret key sk_f and a ciphertext ct, Dec computes and outputs a value $y \in \mathcal{Y}_\lambda$.

Definition 24 (Correctness)**.** Let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be an FE scheme for a function space $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$ with $\#\mathcal{Y} \geq 2$. For a sequence of functions $f = (f_\lambda)_\lambda \in \mathcal{F}$ and a sequence of messages $x = (x_\lambda)_\lambda \in \mathcal{X}$, set

$$p_{f,x}(\mathsf{Dec}) := \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{sk}_f \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_\lambda) \\ \mathsf{ct}_x \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda)}} [\mathsf{Dec}(\mathsf{sk}_f, \mathsf{ct}_x) = f_\lambda(x_\lambda)] \tag{2.31}$$

We define the **decryption probability** of Dec by

$$\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}) := \min_{f \in \mathcal{F}, x \in \mathcal{X}} p_{f,x}(\mathsf{Dec}) \tag{2.32}$$

and the **decryption advantage** of Dec by

$$\mathsf{adv}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}) := \frac{\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}) \cdot \#\mathcal{Y}_\lambda - 1}{\#\mathcal{Y}_\lambda - 1}. \tag{2.33}$$

We call FE **correct** if there is a negligible function $\varepsilon \in \mathsf{negl}(\lambda)$ s.t. we have

$$\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}) \geq 1 - \varepsilon(\lambda). \tag{2.34}$$

Note that $\mathsf{adv}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})$ is normed in the following way: if Dec always outputs the correct message, then its decryption advantage equals 1. However, if Dec outputs uniformly random elements of $\mathcal{Y}$ without looking at the ciphertext, then its decryption advantage is 0. In the worst case, Dec always outputs some fixed element of $\mathcal{Y}$. In this case, Dec's decryption advantage is $\frac{-1}{\#\mathcal{Y}-1}$.

We will introduce secret-key encryption schemes as special cases of functional encryption schemes:

Definition 25 (Secret-Key Encryption)**.** Let $\mathsf{SKE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be an FE scheme for a function space $\mathcal{F} : \mathcal{X} \to \mathcal{X}$. We call SKE a **secret-key encryption** (**SKE**) scheme if each $\mathcal{F}_\lambda$ only contains the identity $\mathsf{id}_{\mathcal{X}_\lambda}$. In this case, we will—without loss of generality—assume that each secret key $\mathsf{sk}_{\mathsf{id}_{\mathcal{X}_\lambda}}$, outputted by $\mathsf{KeyGen}(\mathsf{msk}, \mathsf{id}_{\mathcal{X}_\lambda})$ for some master secret key $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$, is identical to msk, i.e.

$$\mathsf{msk} = \mathsf{sk}_{\mathcal{X}_\lambda}. \tag{2.35}$$

Additionally, we will omit the KeyGen algorithm from the signature of SKE and simply write

$$\mathsf{SKE} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{Dec}) \tag{2.36}$$

instead of $\mathsf{SKE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$.

Note that the notion of correctness that we defined in Definition 23 is also applicable to SKEs.

For technical reasons, we will also introduce the notion of *partial* secret-key encryption schemes. Simply put, a partial SKE scheme is an SKE without a decryption algorithm.

Definition 26 (Partial Secret-Key Encryption)**.** A **partial secret-key encryption** scheme SKE is a pair of two algorithms Setup and Enc s.t. both algorithms have the same syntax as in Definitions 23 and 25. We will denote partial SKEs usually by $\mathsf{SKE} = (\mathsf{Setup}, \mathsf{Enc}, \_)$.

An algorithm Dec is a **fitting decryption algorithm** for the partial scheme $\mathsf{SKE} = (\mathsf{Setup}, \mathsf{Enc}, \_)$ if the triple $(\mathsf{Setup}, \mathsf{Enc}, \mathsf{Dec})$ is an SKE in the sense of Definition 25.

2.1.4 Security Notions

The main security notion that we will study here is the notion of *selective indistinguishability under chosen plaintext attack*, which is usually abbreviated as selective IND-CPA security. This security notion is game-based, i.e., it is defined by a game between a (randomized) challenger and an adversary. In these games, the challenger will draw a random bit $b \leftarrow \{0,1\}$ and hide information about b in ciphertexts of an FE scheme FE. The adversary will get access to ciphertexts and secret keys of FE and will try to extract some non-trivial amount of information from the ciphertexts that will help to correctly guess b. Finally, the adversary will submit its guess to the challenger and will win if its guess turns out to be correct. An FE scheme is secure in the sense of this game if the success probability of each PPT adversary is only by a negligible amount larger than $1/2$.

We will first introduce the selective IND-CPA security game. Afterwards, we will specify when we call an FE scheme selective IND-CPA secure.

Game 3 (Selective IND-CPA Security Game)**.** Let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be an FE scheme for a function space $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$.

We define the **selective IND-CPA security game** of FE as the following game between a stateful challenger $\mathcal{C}$ and a stateful adversary $\mathcal{A}$:

Phase 1: On input 1^λ, the adversary $\mathcal{A}$ sends two lists $(x_i^{(0)})_{i=1}^N, (x_i^{(1)})_{i=1}^N \in \mathcal{X}_\lambda^N$ to the challenger $\mathcal{C}$. Additionally, the adversary sends a list of functions $(f_i)_{i=1}^Q \in \mathcal{F}_\lambda^Q$ to the challenger.

Phase 2: The challenger $\mathcal{C}$ receives as input the unary encoded security parameter 1^λ and collects the lists $(x_i^{(0)})_{i=1}^N, (x_i^{(1)})_{i=1}^N \in \mathcal{X}_\lambda^N$ and $(f_j)_{j=1}^Q \in \mathcal{F}_\lambda^Q$ from the adversary. The challenger draws a random bit $b \leftarrow \{0,1\}$, and samples a fresh master secret key $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$. It encrypts all messages of $(x_i^{(b)})_{i=1}^N$, i.e., it computes for $i = 1, \ldots, N$

$$\mathsf{ct}_i := \mathsf{Enc}(\mathsf{msk}, x_i^{(b)}). \tag{2.37}$$

Further, it generates secret keys for all functions submitted by the adversary, i.e., it computes for $j = 1, \ldots, Q$

$$\mathsf{sk}_j := \mathsf{KeyGen}(\mathsf{msk}, f_j). \tag{2.38}$$

Finally, the challenger sends the list of ciphertexts $(\mathsf{ct}_i)_{i=1}^N$ and the list of secret keys $(\mathsf{sk}_j)_{j=1}^Q$ to the adversary.

Phase 3: Upon receiving $(\mathsf{ct}_i)_{i=1}^N$ and $(\mathsf{sk}_j)_{j=1}^Q$, the adversary $\mathcal{A}$ does some computations on its own and finally responds with a guess $b' \in \{0,1\}$.

The adversary $\mathcal{A}$ **wins** a run of the above game if it guesses the bit b of the challenger correctly, i.e., if $b = b'$, and if it holds for all $i \in [N]$ and $j \in [Q]$

$$f_j(x_i^{(0)}) = f_j(x_i^{(1)}). \tag{2.39}$$

We will call the lengths N and Q of the lists $(x_i^{(0)})_{i=1}^N, (x_i^{(1)})_{i=1}^N$ and $(f_j)_{j=1}^Q$ submitted by the adversary the number of **encryption queries** and **function queries** made by the adversary, respectively.

Note that the requirement in Eq. (2.39) basically states that the adversary is not able to determine b by the trivial amount of information it can gather from the received ciphertexts and secret keys. In fact, if Eq. (2.39) would be violated, let's say $f_j(x_i^{(0)}) \neq f_j(x_i^{(1)})$ for some fixed i, j, then the adversary could determine b by computing $y := \mathsf{Dec}(\mathsf{sk}_j, \mathsf{ct}_i)$. If y equals $f_j(x_i^{(0)})$, then b must be zero (assuming FE is correct), otherwise b must be one.

Definition 27 (Selective IND-CPA Secure FE Schemes)**.** Let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be an FE scheme. We define the **advantage** of an adversary $\mathcal{A}$ against the selective IND-CPA security of FE by

$$\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}) := 2 \cdot \left(\Pr[\mathcal{A} \text{ wins in Game 3}] - \frac{1}{2}\right) \tag{2.40}$$

where the probability is taken over the randomness of $\mathcal{A}$ and $\mathcal{C}$. Note that $\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A})$ is a function in λ.

We say that FE is **selectively IND-CPA secure against a class A** of adversaries if the advantage of each adversary $\mathcal{A} \in \mathbf{A}$ in Game 3 is negligible, i.e., $\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}) \in \mathsf{negl}(\lambda)$. We will call FE **selectively IND-CPA secure** if FE is selectively IND-CPA secure against the class of PPT algorithms.

Since we are not studying other security notions for FE schemes (except of function-hiding security), we will sometimes call an FE or SKE scheme simply *secure* or *IND-CPA secure* when it is selectively IND-CPA secure.

Remark 4 (Stronger Notions of Security)*.* The notion of selective IND-CPA security only guarantees a limited form of security, since it significantly limits its potential interactions of an adversary. In fact, this security notion can be modified in multiple ways. In literature, stronger notions of security are usually proven for FE candidates. We list possible alterations in the following:

1. Game 3 does not allow for a lot of interactions. Specifically, it expects that the adversary $\mathcal{A}$ commits to all messages and functions for which it wants to query ciphertexts and secret keys. Usually, one allows $\mathcal{A}$ to choose messages and functions adaptively, i.e., $\mathcal{A}$ has oracle-access to an encryption oracle and a key generation oracle, which $\mathcal{A}$ may query at any time. Particularly, $\mathcal{A}$ may make its queries dependent on ciphertexts and secret keys it has seen before.

   This leads to the notion of *adaptive IND-CPA security* [ONe10; BSW11].

2. Further, instead of demanding the adversary to distinguish between ciphertexts of two different messages, one could also require the adversary to distinguish between real ciphertexts of the FE scheme and *simulated* ciphertexts, i.e., ciphertexts that are generated by a simulator that does not get to see the actual messages submitted by the adversary. This simulator only knows the values to which the ciphertexts must decrypt for the secret keys queried by the adversary.

   This leads to the notion of *simulation-based security under chosen plaintext attack* (SIM-CPA) [ONe10; BSW11].

3. Additionally, one could permit the adversary to ask the challenger for decryptions of (potentially) malformed ciphertexts. I.e., $\mathcal{A}$ has adaptive access to an oracle that computes $\mathsf{Dec}(\mathsf{msk}, \_)$.

This enhancement of $\mathcal{A}$ leads to the notion of *indistinguishability under chosen ciphertext attack* (IND-CCA) [Lai+22; BBL17].

4. Lastly, note that Game 3 allows *multiple* challenge ciphertexts. There are weaker security games where the adversary is only allowed to submit one pair $(x^{(0)}, x^{(1)})$ of candidate message, while for all other encryption queries it may only submit one message.

   This leads to the notion of *single-challenge security*. However, note that single-challenge and multi-challenge security are equivalent up to polynomial factors. In fact, by a typical hybrid argument one can show that, if there is an adversary with time complexity t that submits N message pairs in Game 3 and has an advantage of $\varepsilon > 0$ against the security of FE, then there is another adversary that has a comparable time-complexity of $O(t + N)$ and an advantage of ε/N in the corresponding single-challenge security game.

Definition 27 does not capture the notion of *function-hiding* security. In fact, Game 3 does not give the adversary the possibility to submit pairs of functions and distinguish between different distributions of secret keys. To define the function-hiding property for FE schemes, we will introduce a second security game:

Game 4 (Selective Function-Hiding Security Game)**.** Let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be an FE scheme for a function space $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$.

We define the **selective function-hiding security game** of FE as the following game between a challenger $\mathcal{C}$ and a stateful adversary $\mathcal{A}$:

Phase 1: On input 1^λ, the adversary $\mathcal{A}$ sends two lists $(f_j^{(0)})_{j=1}^Q, (f_j^{(1)})_{j=1}^Q \in \mathcal{F}_\lambda^Q$ of functions to the challenger $\mathcal{C}$.

Phase 2: The challenger $\mathcal{C}$ receives as input the unary encoded security parameter 1^λ and collects the lists $(f_j^{(0)})_{j=1}^Q, (f_j^{(1)})_{j=1}^Q \in \mathcal{F}_\lambda^Q$ from the adversary. The challenger draws a random bit $b \leftarrow \{0, 1\}$, and samples a fresh master secret key msk $\leftarrow \mathsf{Setup}(1^\lambda)$. It generates secret keys for all functions of the selected list, i.e., it computes for $j = 1, \ldots, Q$

$$\mathsf{sk}_j := \mathsf{KeyGen}(\mathsf{msk}, f_j^{(b)}). \tag{2.41}$$

Finally, the challenger sends the list of secret keys $(\mathsf{sk}_j)_{j=1}^Q$ to the adversary.

Phase 3: Upon receiving $(\mathsf{sk}_j)_{j=1}^Q$, the adversary $\mathcal{A}$ does some computations on its own and finally responds with a guess $b' \in \{0, 1\}$.

The adversary $\mathcal{A}$ **wins** a run of the above game if it guesses the bit b of the challenger correctly, i.e., $b = b'$. We will call the length Q of the lists $(f_j^{(0)})_{j=1}^Q, (f_j^{(1)})_{j=1}^Q$ submitted by the adversary the number of **function queries** made by the adversary.

Definition 28 (Selective IND-CPA Function-Hiding Secure FE Schemes)**.** Let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be an FE scheme. We define the **advantage** of an adversary $\mathcal{A}$ against the selective function-hiding security of FE by

$$\mathsf{adv}_{\mathsf{FE}}^{\mathsf{FH}}(\mathcal{A}) := 2 \cdot \left(\Pr[\mathcal{A} \text{ wins in Game 4}] - \frac{1}{2}\right) \tag{2.42}$$

where the probability is taken over the randomness of $\mathcal{A}$ and $\mathcal{C}$. Note that $\mathsf{adv}_{\mathsf{FE}}^{\mathsf{FH}}(\mathcal{A})$ is a function in λ.

We say that FE is **selectively function-hiding secure against a class $\mathbf{A}$** of adversaries if the advantage of each adversary $\mathcal{A} \in \mathbf{A}$ in Game 4 is negligible, i.e., $\mathsf{adv}_{\mathsf{FE}}^{\mathsf{FH}}(\mathcal{A}) \in \mathsf{negl}(\lambda)$. We will call FE **selectively function-hiding secure** if FE is selectively function-hiding secure against the class of PPT adversaries.

Finally, we will call FE **selectively function-hiding IND-CPA secure against the class $\mathbf{A}$** if FE is selectively function-hiding and IND-CPA secure against $\mathbf{A}$. If $\mathbf{A}$ is the class of PPT adversaries, we will simply call FE **selectively function-hiding IND-CPA secure**.

Again, since we do not consider other notions of security here besides selective IND-CPA security, we will call FE simply *function-hiding secure* or just *function-hiding* if it is selectively function-hiding IND-CPA secure.

Remark 5 (Stronger Notions of Function-Hiding Security). Our definition of selective function-hiding IND-CPA security is very weak. In the literature [BRS13a; BJK15; Agr+15; BS15], one considers a combination of Game 3 and Game 4 where the adversary submits two lists $(x_i^{(0)})_{i=1}^N, (x_i^{(1)})_{i=1}^N$ of candidate messages *and* two lists $(f_j^{(0)})_{j=1}^Q, (f_j^{(1)})_{j=1}^Q$ of candidate functions. For a random bit $b \leftarrow \{0,1\}$, the adversary then receives the ciphertexts of $(x_i^{(b)})_{i=1}^N$ and the secret keys of $(f_j^{(b)})_{j=1}^Q$, and has to guess b. For this decisional game to be fair, one requires the adversary to only submit pairs of candidate message and candidate functions such that, for all $i = 1, \ldots, N$ and $j = 1, \ldots, Q$,

$$f_j^{(0)}(x_i^{(0)}) = f_j^{(0)}(x_i^{(1)}) = f_j^{(1)}(x_i^{(0)}) = f_j^{(1)}(x_i^{(1)}). \tag{2.43}$$

There are even stronger notions of function-hiding security where one only demands

$$f_j^{(0)}(x_i^{(0)}) = f_j^{(1)}(x_i^{(1)}) \tag{2.44}$$

for all $j \in [Q]$ and $i \in [N]$.

2.1.5 Lattice-Based Schemes

In this subsection, we will specify when we call an FE scheme *lattice-based.* We require that in a lattice-based FE scheme the circuit complexity of the encryption and decryption algorithms are limited: we will demand that the encryption algorithm Enc can be separated in an offline and an online phase where the offline phase of Enc may be computationally unbounded, but does not get to see the message that is to be encrypted. In its online phase, Enc gets the intermediate results of its offline phase and sees the message, however it is now limited to evaluate an arithmetic circuit of constant depth.

For the decryption procedure of a lattice-based FE scheme, we will require that it applies a polynomial of constant degree to a given ciphertext and secret key and rounds it to the nearest value in the value space.

Formally, we define for an encryption algorithm:

Definition 29 (Encryption Algorithms of Limited Depth and Width)**.** Let Enc be the encryption algorithm of an FE scheme $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$

or of a partial SKE scheme $\mathsf{SKE} = (\mathsf{Setup}, \mathsf{Enc}, \_)$ with message space $\mathcal{X} \subset \mathbb{Z}^n$. Further, let R be either $\mathbb{Z}$ or $\mathbb{Z}_q$ for some $q = q(\lambda) \in \mathbb{N}$ and assume that each ciphertext output by Enc is an element in R^m for some $m \in \mathsf{poly}(\lambda)$.

We will introduce the following notions:

1. We say that Enc is of **width** B over the ring R if there is a bound[1] $B > 0$ and a negligible function $\varepsilon \in \mathsf{negl}(\lambda)$ s.t. we have for each sequence $(x_\lambda)_\lambda \in \mathcal{X}$
$$\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda)}} [\ ||\mathsf{ct}||_\infty > B] \leq \varepsilon(\lambda). \tag{2.45}$$

2. We say that Enc is of **depth** d over R if there is an algorithm $\mathsf{Enc}_{\mathsf{off}}$ s.t. for each $\lambda \in \mathbb{N}$, each $x \in \mathcal{X}_\lambda$ and each $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$ the following two things hold:
   (a) On input msk, $\mathsf{Enc}_{\mathsf{off}}$ will always output m polynomials $r_1, \ldots, r_m \in R[X_1, \ldots, X_n]$ of degree $\leq d$.
   (b) The output distribution of $\mathsf{Enc}(\mathsf{msk}, x)$ is identical to the output of the following subroutine:
```
1: $(r_1, \ldots, r_m) \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$
2: $\mathsf{ct}_x := (r_1(x), \ldots, r_m(x))$
3: return $\mathsf{ct}_x$
```
   We will call $\mathsf{Enc}_{\mathsf{off}}$ the **offline part** of Enc. The **online part** of Enc is given by evaluating $(r_1, \ldots, r_m)$ at input x.

For encryption algorithms of constant depth, we have the following lemma:

Lemma 52. *For* $\mathsf{SKE} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{Dec})$ *of depth d with $\#\mathcal{X} \geq 2$, we have*
$$\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ (r_1, \ldots, r_m) \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})}} [\exists j \in [m] : \deg r_j > 0] \geq \mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}). \tag{2.46}$$

Proof. Draw $x \leftarrow \mathcal{X}_\lambda$ uniformly at random, and sample $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$ and $r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$. Denote the event that each r_j is constant by $\deg r = 0$. Set
$$\gamma := \Pr[\deg r \neq 0] = \Pr[\exists j \in [m] : \deg r_j > 0]. \tag{2.47}$$

We assume—without loss of generality— that $0 \in \mathcal{X}$. Note that we have $r(x) = r(0)$ whenever $\deg r = 0$. Let us consider the following inequality
$$\Pr[\mathsf{Dec}(\mathsf{msk}, r(x)) = x] \tag{2.48}$$
$$= \Pr[\mathsf{Dec}(\mathsf{msk}, r(x)) = x \mid \deg r \neq 0] \cdot \Pr[\deg r \neq 0] \tag{2.49}$$
$$+ \Pr[\mathsf{Dec}(\mathsf{msk}, r(x)) = x \mid \deg r = 0] \cdot \Pr[\deg r = 0] \tag{2.50}$$
$$= \Pr[\mathsf{Dec}(\mathsf{msk}, r(x)) = x \mid \deg r \neq 0] \cdot \gamma \tag{2.51}$$
$$+ \Pr[\mathsf{Dec}(\mathsf{msk}, r(x)) = x \mid \deg r = 0] \cdot (1 - \gamma) \tag{2.52}$$
$$\leq \gamma + (1 - \gamma) \cdot \Pr[\mathsf{Dec}(\mathsf{msk}, r(0)) = x \mid \deg r = 0]. \tag{2.53}$$

[1] Remember that we established in Section 2.1.1 to associate $||\_ \bmod q||_\infty$, the infinity norm modulo q, with $\mathbb{Z}_q$ and to write $||\mathsf{ct}||_\infty$ instead of $||\mathsf{ct} \bmod q||_\infty$ whenever $\mathsf{ct} \in \mathbb{Z}_q^m$.

The distribution $\mathsf{Dec}(\mathsf{msk}, r(0))$ is independent of x (even when conditioned on $\deg r = 0$). Since x has been chosen uniformly at random from $\mathcal{X}_\lambda$, it follows $\Pr[\mathsf{Dec}(\mathsf{msk}, r(0)) = x \mid \deg r = 0] = 1/\#\mathcal{X}_\lambda$. Hence, we have

$$\begin{aligned} &\Pr[\mathsf{Dec}(\mathsf{msk}, r(x)) = x] && (2.54)\\ \leq& \gamma + (1-\gamma) \cdot \Pr[\mathsf{Dec}(\mathsf{msk}, r(0)) = x \mid \deg r = 0] && (2.55)\\ =& \gamma + \frac{1-\gamma}{\#\mathcal{X}_\lambda}. && (2.56) \end{aligned}$$

For $\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec})$, which was the minimum of $\Pr[\mathsf{Dec}(\mathsf{msk}, r(x)) = x]$ for each x, it now follows

$$\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}) \leq \Pr_{x \leftarrow \mathcal{X}_\lambda}[\mathsf{Dec}(\mathsf{msk}, r(x)) = x] \leq \gamma + \frac{1-\gamma}{\#\mathcal{X}_\lambda}, \tag{2.57}$$

which is equivalent to the inequality $\gamma \geq \frac{\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}) \cdot \#\mathcal{X}_\lambda - 1}{\#\mathcal{X}_\lambda - 1} = \mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec})$. □

Definition 30 (Lattice-Based FE)**.** Let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be an FE scheme for a function space $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$. Let $d_1, d_2 \in \mathbb{N}$ be constant and let $q, m, B \in \mathbb{N}$ with $B < q/2$. We say that FE is **lattice-based** if the following conditions are met:

1. Each ciphertext output by Enc lies in $\mathbb{Z}_q^m$.
2. Each secret key output by KeyGen is a polynomial in $\mathbb{Z}_q[C_1, \ldots, C_m]$.
3. Enc is of depth d_1 over $\mathbb{Z}_q$.
4. For each sequence of messages $(x_\lambda)_\lambda \in \mathcal{X}$ and for each sequence of functions $(f_\lambda)_\lambda \in \mathcal{F}$ we have for all $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda), \mathsf{sk}_f \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_\lambda)$ and $\mathsf{ct}_x \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda)$
$$\mathsf{Dec}(\mathsf{sk}_f, \mathsf{ct}_x) = 0 \iff |\mathsf{sk}(\mathsf{ct}) \bmod q| < B. \tag{2.58}$$
5. The output of $\mathsf{Dec}(\mathsf{sk}_f, \mathsf{ct}_x)$ only depends on the evaluation $\mathsf{sk}_f(\mathsf{ct}_x) \in \mathbb{Z}_q$.

We will call d_1 the **encryption depth**, d_2 the **decryption depth** and B the **noise bound** of FE.

The last two requirements of Definition 30 may seem strange; however they are a natural weakening of requiring that Dec works by computing $\mathsf{sk}_f(\mathsf{ct}_x)$ and rounding it to the nearest value in $\mathcal{Y}$. In fact, if we have $\mathcal{Y} = \mathbb{Z}_p$, for $p < \frac{q}{2}$, and if decryption is given by

$$\mathsf{Dec}(\mathsf{sk}_f, \mathsf{ct}_x) = \left\lceil \frac{\mathsf{sk}_f(\mathsf{ct}_x)}{\lceil q/p \rceil} \right\rfloor, \tag{2.59}$$

then Eq. (2.58) directly follows for the noise bound $B = \frac{q}{2p}$.

Some lattice-based schemes use a different style of decryption. Instead of computing $\left\lceil \frac{\mathsf{sk}_f(\mathsf{ct}_x)}{\lceil q/p \rceil} \right\rfloor$, the decryption algorithm computes $\mathsf{sk}_f(\mathsf{ct}_x) \bmod p$, where it interprets the field element $\mathsf{sk}_f(\mathsf{ct}_x) \in \mathbb{Z}_q$ as a number in $\left\{-\frac{q-1}{2}, \ldots, \frac{q-1}{2}\right\}$ and computes the corresponding remainder in $\left\{-\frac{p-1}{2}, \ldots, \frac{p-1}{2}\right\}$ when dividing $\mathsf{sk}_f(\mathsf{ct}_x) \in \left\{-\frac{q-1}{2}, \ldots, \frac{q-1}{2}\right\}$ by p. As far as it concerns our lower bounds, this is not a significant difference. Indeed, the following lemma shows that rounding from $\mathbb{Z}_q$ to $\mathbb{Z}_p$ is—up to scaling with the multiplicative inverse of p modulo q—approximately close to arithmetically reducing from $\mathbb{Z}_q$ to $\mathbb{Z}_p$.

Lemma 53. *Let $p, q \in \mathbb{N}$ be odd, coprime numbers with $p < q$ and let $x \in \left\{-\frac{q-1}{2}, \ldots, \frac{q-1}{2}\right\} \subset \mathbb{Z}$. For $y \in \left\{-\frac{q-1}{2}, \ldots, \frac{q-1}{2}\right\}$ s.t. $yp \bmod q = 1$ We have*

$$|(y \cdot (x \bmod p) - y \cdot x) \bmod q| < \frac{q}{2p} \tag{2.60}$$

where $x \bmod p \in \left\{-\frac{p-1}{2}, \ldots, \frac{p-1}{2}\right\}$.

In particular, if $x \bmod p = 0$, it follows

$$|yx \bmod q| < \frac{q}{2p}. \tag{2.61}$$

Proof. Write x as

$$x = a \cdot p + b \tag{2.62}$$

with a, b unique s.t. $b \in \left\{-\frac{p-1}{2}, \ldots, \frac{p-1}{2}\right\}$. It must then follow $|x - b| < q/2$. Hence, we have for the absolute values of real numbers

$$|a| = \left|\frac{x-b}{p}\right| = \frac{|x-b|}{p} < \frac{q/2}{p} = \frac{q}{2p}. \tag{2.63}$$

Further, we have

$$(y \cdot (x \bmod p) - y \cdot x) = y \cdot b - y \cdot (a \cdot p + b) = yp \cdot a. \tag{2.64}$$

With $(yp \cdot a) \bmod q = a$, Eq. (2.60) follows. □

Unfortunately, it is not clear if Lemma 53 can be transferred to FE schemes that apply multiple arithmetic reductions at decryption. The quadratic FE scheme of Agrawal and Rosen [AR17], for example, uses three primes $p_1 < p_2 < q$ and decrypts by computing

$$((\mathsf{sk}_f(\mathsf{ct}_x) \bmod q) \bmod p_2) \bmod p_1. \tag{2.65}$$

In this situation, it is not clear if we can scale $\mathsf{sk}_f(\mathsf{ct}_x) \in \mathbb{Z}_q$ with some value $\alpha \in \mathbb{Z}_q$ s.t. $|\alpha \cdot \mathsf{sk}_f(\mathsf{ct}_x) \bmod q|$ is bounded whenever $f(x) = 0$.

2.2 Multivariate Interpolation

To prove the results of Section 2.4, we will make use of polynomial *interpolation*, i.e., the process of deducing the coefficients of a polynomial from its evaluations on certain points. A popular tool for this is the *Vandermonde* matrix. In the univariate degree-d case, the Vandermonde matrix for $d+1$ points $x_1, \ldots, x_{d+1} \in \mathbb{R}$ is given by

$$V(x_1, \ldots, x_{d+1}) = \begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^d \\ 1 & x_2 & x_2^2 & \cdots & x_2^d \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{d+1} & x_{d+1}^2 & \cdots & x_{d+1}^d \end{pmatrix}. \tag{2.66}$$

If $\mathrm{coeff}(f)$ is the coefficient vector of a degree-d polynomial f, then $V(x_1, \ldots, x_{d+1})$ maps $\mathrm{coeff}(f)$ to its evaluations $(f(x_1), \ldots, f(x_{d+1}))$. Hence, f can be interpolated by applying $V(x_1, \ldots, x_{d+1})^{-1}$ to $(f(x_1), \ldots, f(x_{d+1}))$. The univariate

Vandermonde matrix is well-studied: we know explicit formulas to compute its determinant and its inverse.

By generalizing the concept of the univariate Vandermonde matrix, one can deduce the *multivariate* Vandermonde matrix. The multivariate Vandermonde matrix maps linearly the coefficients of a degree-d polynomial f over n variables to the values of f on $\binom{n+d}{d}$ interpolation points. Compared to the univariate case, the multivariate Vandermonde matrix has been studied less extensively. For example, there is no closed formula known for calculating the determinant of the multivariate Vandermonde matrix [Olv06; DU14; Ben14].

In cryptography, Vandermonde matrices have multiple applications beside cryptanalysis. For example, in the setting of ideal lattices, one can use Vandermonde matrices to extract witnesses from proof systems [AL21]. Hence, Vandermonde matrices are of independent interest. We will spend this section to study the multivariate Vandermonde matrix over a special set of interpolation points.

Before we define the Vandermonde matrix for univariate and multivariate integer polynomials, we will first establish a convention for this section:

Convention 2. For each $n \in \mathbb{N}$, fix in this section some bijective map $\alpha^{(n)} : \mathbb{N} \to \mathbb{N}_0^n$ s.t. we have for all $i, j \in \mathbb{N}$

$$i < j \iff \left|\left|\alpha^{(n)}(i)\right|\right|_1 \leq \left|\left|\alpha^{(n)}(j)\right|\right|_1. \tag{2.67}$$

Then, $\alpha^{(n)}$ induces a degree ordering on the set of all monomials of $\mathbb{Z}[X] = \mathbb{Z}[X_1, \ldots, \mathcal{X}_n]$ and for each $d \in \mathbb{N}_0$, the list

$$(X^{\alpha^{(n)}(1)}, \ldots, X^{\alpha^{(n)}(\binom{n+d}{d})}) \tag{2.68}$$

enumerates all monomials of $\mathbb{Z}[X]$ of total degree $\leq d$.

Definition 31 (Vandermonde Matrix)**.** Let $n, d \in \mathbb{N}$ and set $L := \binom{n+d}{d}$. Choose L points $x_1, \ldots, x_L \in \mathbb{Z}^n$. The **Vandermonde matrix** with respect to $x_1, \ldots, x_L$ is given by

$$V(x_1, \ldots, x_L) := \begin{pmatrix} x_1^{\alpha^{(n)}(1)} & x_1^{\alpha^{(n)}(2)} & \ldots & x_1^{\alpha^{(n)}(L)} \\ x_2^{\alpha^{(n)}(1)} & x_2^{\alpha^{(n)}(2)} & \ldots & x_2^{\alpha^{(n)}(L)} \\ \vdots & \vdots & \ddots & \vdots \\ x_L^{\alpha^{(n)}(1)} & x_L^{\alpha^{(n)}(2)} & \ldots & x_L^{\alpha^{(n)}(L)} \end{pmatrix} \tag{2.69}$$

where for $x_i = (x_{i,1}, \ldots, x_{i,n})$ and $\alpha^{(n)}(j) = (\alpha_1^{(n)}(j), \ldots, \alpha_n^{(n)}(j))$ the (i,j)-th entry is given by

$$x_i^{\alpha^{(n)}(j)} = x_{i,1}^{\alpha_1^{(n)}(j)} \cdots x_{i,n}^{\alpha_n^{(n)}(j)}. \tag{2.70}$$

We will denote by $V(n,d)$ the Vandermonde matrix where $x_i = \alpha^{(n)}(i)$, i.e.,

$$V(n,d) := V(\alpha(1), \ldots, \alpha(L)). \tag{2.71}$$

For an integer polynomial $f(X) = \sum_{i=1}^{L} c_i \cdot X^{\alpha^{(n)}(i)}$ of degree $\leq d$ (where L is still $\binom{n+d}{d}$)), denote its column vector of coefficients by

$$\mathrm{coeff}_d(p) := \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_L \end{pmatrix}. \tag{2.72}$$

We then have

$$V(x_1, \ldots, x_L) \cdot \mathrm{coeff}_d(p) \tag{2.73}$$

$$= \begin{pmatrix} x_1^{\alpha^{(n)}(1)} & x_1^{\alpha^{(n)}(2)} & \ldots & x_1^{\alpha^{(n)}(L)} \\ x_2^{\alpha^{(n)}(1)} & x_2^{\alpha^{(n)}(2)} & \ldots & x_2^{\alpha^{(n)}(L)} \\ \vdots & \vdots & \ddots & \vdots \\ x_L^{\alpha^{(n)}(1)} & x_L^{\alpha^{(n)}(2)} & \ldots & x_L^{\alpha^{(n)}(L)} \end{pmatrix} \cdot \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_L \end{pmatrix} = \begin{pmatrix} p(x_1) \\ p(x_2) \\ \vdots \\ p(x_L) \end{pmatrix}. \tag{2.74}$$

This means, the Vandermonde matrix with respect to the points $x_1, \ldots, x_L$ maps the coefficients of a polynomial f linearly to its evaluations $f(x_1), \ldots, f(x_L)$ on $x_1, \ldots, x_L$. By reverting this process, we can extract the coefficients of f from its values $f(x_1), \ldots, f(x_L)$. However, for this to work, it is necessary that $V(x_1, \ldots, x_L)$ is invertible. Additionally, in the case of lattice-based cryptography, it does not suffice that $V(x_1, \ldots, x_L)^{-1}$ exists. We also need that $V(x_1, \ldots, x_L)^{-1}$ behaves well and, ideally, has small entries. This leads to the following problem: assume that $d \in \mathbb{N}$ is constant and let n be an arbitrary parameter. Further, let q be a prime modulus. The entries of the Vandermonde matrix $V(n, d)$ for our special set of interpolation points from Definition 31 are bounded by d^d and, hence, lie in $O(1)$. Further, one can show that $V(n, d)$ is invertible over $\mathbb{Z}_q$ if $q > d$. However, the inverse of $V(n, d)$ over $\mathbb{Z}_q$ does not need to have small entries. In fact, the entries of the inversion over $\mathbb{Z}_q$ will lie in $\Theta(q)$.

We can solve this problem as follows: if we set $W(n, d) := d! \cdot V(n, d)^{-1}$ (where $V(n, d)^{-1}$ denotes the inverse over $\mathbb{R}$), then $W(n, d)$ is—as we will show—integer. Further, the infinity norm of $W(n, d)$ is bounded by $d! \cdot (2d)!$. It follows that the infinity norm of $W(n, d) \bmod q$ is constant, too, and we have

$$W(n, d) \cdot V(n, d) = (d!) \cdot \mathsf{id}_{L \times L} \mod q. \tag{2.75}$$

We will call $W(n, d)$ a *quasi-inverse*, since it behaves up to the scalar $d!$ like the real inverse $V(n, d)^{-1}$, however it is integer and retains its good properties when we cast it modulo q.

Note that the bounds and qualities of $W(n, d)$ grow exponentially in d, however, are independent of n. This means, while incrementing the degree of polynomials makes interpolation costlier, for our special set of interpolation points, we can increase the number of variables without deteriorating the quality of our interpolation method in the lattice-based setting.

We will summarize our observations and results in the main theorem of this section:

Theorem 54 (Multivariate Interpolation). *1. For $n, d \in \mathbb{N}$, we have*

$$\det V(n,d) = \prod_{i=1}^{d} (d+1-i)^{i \cdot \binom{n-1+i}{i}}. \tag{2.76}$$

2. For $n, d \in \mathbb{N}$, set $W(n,d) := d! \cdot V(n,d)^{-1}$. Then, $W(n,d)$ lies in $\mathbb{Z}^{L \times L}$ where $L = \binom{n+d}{d}$.

*3. We define the **interpolation number** Γ_d for degree $d \in \mathbb{N}$ by*

$$\Gamma_d := \max_{n \in \mathbb{N}} \left|\left|V(n,d)^{-1}\right|\right|_\infty. \tag{2.77}$$

Γ_d is a well-defined element of $\mathbb{N}$, and we have

$$\Gamma_d \leq (2d)!. \tag{2.78}$$

As a direct application of Theorem 54, we will prove the following theorem about distributions of random polynomials, which we will need for Section 2.4:

Theorem 55. *Let $\mathcal{D}$ be a distribution with support in $\mathbb{Z}[X_1]^{\leq d}$. Let $B \geq 1/2$ s.t. we have for each $f \leftarrow \mathcal{D}$ and $x \in \{0, \ldots, d\}$*

$$|f(x)| \leq B. \tag{2.79}$$

Further, let $e > 0$ s.t. we have for each $x \in [2d]$

$$\left| \mathop{\mathbb{E}}_{f \leftarrow \mathcal{D}} [f(x)^2 - f(0)^2] \right| \leq e. \tag{2.80}$$

For $i \in [d]$, we have

$$\Pr[\deg f > d - i] \leq \Gamma_{2d} \cdot e \cdot (1 + 2B \cdot \Gamma_d)^i. \tag{2.81}$$

We will spend Section 2.2.1 on proving Theorem 54. In Section 2.2.2, we will discuss the tightness of the bounds on $W(n,d)$ and Γ_d obtained here. Finally, we will use our findings to prove Theorem 55 in Section 2.2.3.

2.2.1 Proving Theorem 54

We will first prove the second and third part of Theorem 54, which are of greater interest for us. Afterwards, we will thoroughly prove the first part of Theorem 54.

Note that the infinity norm of $V(n,d)^{-1}$ and $W(n,d)$ are related by

$$||W(n,d)||_\infty = \left|\left|d! \cdot V(n,d)^{-1}\right|\right|_\infty = d! \cdot \left|\left|V(n,d)^{-1}\right|\right|_\infty. \tag{2.82}$$

Hence, instead of bounding $\left|\left|V(n,d)^{-1}\right|\right|_\infty$, we will in the following bound $||W(n,d)||_\infty$ by $d! \cdot (2d)!$, which will imply that Γ_d is bounded from above by $(2d)!$. For this end, we will first observe that for $n \geq d$, $||V(n,d)||_\infty$ and $||W(n,d)||_\infty$ do not grow any more in n.

Lemma 56. *Let $n, d \in \mathbb{N}$ with $n > d$. We have*

$$||V(n,d)||_\infty \leq ||V(n-1,d)||_\infty, \tag{2.83}$$

$$||W(n,d)||_\infty \leq ||W(n-1,d)||_\infty. \tag{2.84}$$

Proof. Set $L := \binom{n+d}{d}$ and $L' := \binom{n-1+d}{d}$, and note that the infinity norm of a matrix equals the maximum among sums of absolute entries of each row. Hence, it suffices to show that for each row of $V(n,d)$ resp. $W(n,d)$ there is a row in $V(n-1,d)$ resp. $W(n-1,d)$ whose sum of absolute entries is not smaller. Let $i \in [L]$, and consider the vector $\alpha^{(n)}(i)$, which corresponds to the i-th row x of $V(n,d)$, which is given by

$$x := \left(\alpha^{(n)}(i)^{\alpha^{(n)}(1)}, \alpha^{(n)}(i)^{\alpha^{(n)}(2)}, \ldots, \alpha^{(n)}(i)^{\alpha^{(n)}(L)}\right) \in \mathbb{Z}^L. \tag{2.85}$$

Since $n > d$ and $\left|\left|\alpha^{(n)}(i)\right|\right|_1 \leq d$, one entry of $\alpha^{(n)}(i)$ must be zero. W.l.o.g., the last entry of $\alpha^{(n)}(i)$ is zero. Hence, there must be a $j \in [L']$ s.t. $\alpha^{(n)}(i) = (\alpha^{(n-1)}(j), 0)$. Now, $\alpha^{(n-1)}(j)$ corresponds to the j-th row y of $V(n-1,d)$, which is given by

$$y := \left(\alpha^{(n-1)}(j)^{\alpha^{(n-1)}(1)}, \alpha^{(n-1)}(j)^{\alpha^{(n-1)}(2)}, \ldots, \alpha^{(n-1)}(j)^{\alpha^{(n-1)}(L')}\right). \tag{2.86}$$

For each entry $\alpha^{(n-1)}(j)^{\alpha^{(n-1)}(k)}$ of y there is exactly one corresponding entry $(\alpha^{(n-1)}(j), 0)^{(\alpha^{(n-1)}(k),0)}$ of x that has the same value. On the other hand, each entry $\alpha^{(n)}(i)^{\alpha^{(n)}(k)}$ of x that does not correspond to an entry of y must be zero, since the last entry of $\alpha^{(n)}(k)$ cannot be zero. It follows $||x||_1 = ||y||_1$, hence, for each row of $V(n,d)$ there is one row in $V(n-1,d)$ that has the same one-norm.

To study $||W(n,d)||_\infty$, we will first take a look at $V(n,d)^T$. The transposed matrix $V(n,d)^T$ maps a vector $c \in \mathbb{R}^L$ to the values

$$V(n,d)^T \cdot c = \begin{pmatrix} c_1 \cdot \alpha^{(n)}(1)^{\alpha^{(n)}(1)} + \ldots + c_L \cdot \alpha^{(n)}(L)^{\alpha^{(n)}(1)} \\ c_1 \cdot \alpha^{(n)}(1)^{\alpha^{(n)}(2)} + \ldots + c_L \cdot \alpha^{(n)}(L)^{\alpha^{(n)}(2)} \\ \vdots \\ c_1 \cdot \alpha^{(n)}(1)^{\alpha^{(n)}(L)} + \ldots + c_L \cdot \alpha^{(n)}(L)^{\alpha^{(n)}(L)} \end{pmatrix}. \tag{2.87}$$

This gives rise to an interesting space of exponential functions

$$T[X]^{\leq d} := T[X_1, \ldots, X_n]^{\leq d} := \tag{2.88}$$

$$\left\{ f : \mathbb{N}_0^n \to \mathbb{R} \;\middle|\; \exists c \in \mathbb{R}^L : \; f(X) = c_1 \cdot \alpha^{(n)}(1)^X + \ldots + c_L \cdot \alpha^{(n)}(L)^X \right\}. \tag{2.89}$$

Since $\det V(n,d) \neq 0$ (as we will see later), $V(n,d)^T$ is regular. Hence, a function $f \in T[X]^{\leq d}$ is uniquely determined by its values $f(\alpha^{(n)}(1)), \ldots, f(\alpha^{(n)}(L))$.

Now, let us consider the i-th row v of $W(n,d)$. The vector v induces a function $f_n : \mathbb{N}_0^n \to \mathbb{R}$

$$f_n(X) := v_1 \cdot \alpha^{(n)}(1)^X + \ldots + v_L \cdot \alpha^{(n)}(L)^X. \tag{2.90}$$

Since $W(n,d) \cdot V(n,d) = d! \cdot \mathsf{id}_{n \times n}$, we have for each $j \in [L]$

$$f_n\left(\alpha^{(n)}(j)\right) = \begin{cases} d!, & \text{if } j = i, \\ 0, & \text{if } j \neq i. \end{cases} \tag{2.91}$$

Since $n > d$, one entry of $\alpha^{(n)}(i)$ must be zero. Without loss of generality, we can assume that the last entry of $\alpha^{(n)}(i)$ is zero. Then, there is a $k \in [L']$ s.t. $(\alpha^{(n-1)}(k), 0) = \alpha^{(n)}(i)$. The k-th row w of $W(n-1,d)$ induces a function $f_{n-1} \in T[X_1, \ldots, X_{n-1}]^{\leq d}$ s.t. for each $j \in [L']$

$$f_{n-1}\Big(\alpha^{(n-1)}(j)\Big) = \begin{cases} d!, & \text{if } j = k, \\ 0, & \text{if } j \neq k. \end{cases} \tag{2.92}$$

Denote by $w_1, \ldots, w_{L'}$ the coefficients of f_{n-1} and let $d' = d - \left|\left|\alpha^{(n)}(i)\right|\right|_1$. We define a function $h \in T[X]^{\leq d}$ by

$$\begin{aligned} &h(X_1, \ldots, X_{n-1}, X_n) := f_{n-1}(X_1, \ldots, X_{n-1}) \cdot 0^{X_n} \\ =&w_1 \cdot \alpha^{(n-1)}(1)^{(X_1,\ldots,X_{n-1})} \cdot 0^{X_n} + \ldots + w_{L'} \cdot \alpha^{(n-1)}(L')^{(X_1,\ldots,X_{n-1})} \cdot 0^{X_n} \\ =&w_1 \cdot (\alpha^{(n-1)}(1), 0)^X + \ldots + w_{L'} \cdot (\alpha^{(n-1)}(L'), 0)^X. \end{aligned}$$

Note that we have $h(X_1, \ldots, X_{n-1}, 0) = f_{n-1}(X_1, \ldots, X_{n-1})$. We claim that h equals f_n. Let $j \in [L]$ s.t. the last coordinate of $\alpha^{(n)}(j)$ is zero. If $j \neq i$, we have

$$h(\alpha^{(n)}(j)) = f_{n-1}(\alpha^{(n-1)}(l)) = 0 = f_n(\alpha^{(n)}(j)) \tag{2.93}$$

where $l \in [L']$ s.t. $\alpha^n(j) = (\alpha^{(n-1)}(l), 0)$. For $j = i$, we have

$$h(\alpha^{(n)}(i)) = f_{n-1}(\alpha^{(n-1)}(k)) = d! = f_n(\alpha^{(n)}(i)). \tag{2.94}$$

Now, let $j \in [L]$ s.t. the last coordinate of $\alpha^{(n)}(j) =: (\alpha^{(n-1)}(l), y)$ is not zero. We then have

$$h(\alpha^{(n)}(j)) = f_{n-1}(\alpha^{(n-1)}(l)) \cdot 0^y = 0 = f_n(\alpha^{(n)}(j)). \tag{2.95}$$

Hence, h and f_n coincide on the points $\alpha^{(n)}(1), \ldots, \alpha^{(n)}(L)$ and are, therefore, equal. In particular, h and f_n have the same coefficient vectors. Since $h(X) = 0^{X_n} \cdot f_{n-1}(X)$, the non-zero coefficients of h equal the non-zero coefficients of f_{n-1}. Since f_n corresponds to the i-th row v of $W(n,d)$ and f_{n-1} corresponds to the k-th row w of $W(n,d)$, it follows $||v||_1 = ||w||_1$. In particular, we have for $n > d$

$$||W(n,d)||_\infty \leq ||W(n-1,d)||_\infty. \qquad \square$$

Looking ahead, we will prove in Lemma 64 that $V(n,d)$ admits (up to permutation of rows and columns with a permutation matrix P) a nice block decomposition

$$P \cdot V(n,d) \cdot P^{-1} = \begin{pmatrix} V(n-1,d) & 0 \\ * & V(n,d-1) \end{pmatrix}. \tag{2.96}$$

From this decomposition, it follows that $||V(n-1,d)||_\infty \leq ||V(n,d)||_\infty$ for all $n, d \in \mathbb{N}$. Further, $V(n,d)^{-1}$, and therefore $W(n,d)$ must admit similar decompositions (up to permutation with P). Analogously, we get $||W(n-1,d)||_\infty \leq ||W(n,d)||_\infty$. We can summarize these observations as follows:

Proposition 57. *For $n, d \in \mathbb{N}$, we have*

$$||V(n-1,d)||_\infty \leq ||V(n,d)||_\infty, \tag{2.97}$$
$$||W(n-1,d)||_\infty \leq ||W(n,d)||_\infty. \tag{2.98}$$

Lemma 58. *Let $l_1, \ldots, l_d \in \mathbb{Z}[X_1, \ldots, X_n]$ be degree-1 polynomials. Additionally, let there be numbers $c_1, \ldots, c_d > 0$ s.t. the absolute value of each coefficient of l_i is bounded by c_i.*

If we write $f(X) := l_1(X) \cdots l_d(X)$ as

$$f(X) = \sum_{i=1}^{L} c'_i X^{\alpha^{(n)}(i)}, \tag{2.99}$$

then we have $|c'_i| \leq d! \cdot c_1 \cdots c_d$ for each $i \in [L]$.

Proof. For $k \in [d]$, set

$$\overline{l_k}(X) := c_k X_1 + \ldots + c_k X_n. \tag{2.100}$$

Then, the absolute value of each coefficient of l_k is upper bound by the corresponding coefficient of $\overline{l_k}$. Since all coefficients of $\overline{l_k}$ are positive, it, in particular, follows that the absolute value of each coefficient of $l_j \cdot l_k$ is upper bound by the respective coefficient of $\overline{l_j} \cdot \overline{l_k}$. Inductively, it follows that the absolute value of each coefficient of $f = l_1 \cdots l_d$ is upper bound by the corresponding coefficient of $\overline{f} := \overline{l_1} \cdots \overline{l_d}$.

For $\overline{f}$, we have

$$\overline{f}(X) = \overline{l_1}(X) \cdots \overline{l_d}(X) \tag{2.101}$$
$$= (c_1 X_1 + \ldots + c_1 X_n) \cdots (c_d X_1 + \ldots + c_d X_n) \tag{2.102}$$
$$= c_1 \cdots c_d \cdot (X_1 + \ldots + X_n)^d. \tag{2.103}$$

To finish the proof we will show—by induction on $d \in \mathbb{N}$—that each coefficient of $(X_1 + \ldots + X_n)^d$ is bounded by $d!$. Denote by e_i the i-th unit vector for $i = 1, \ldots, n$, and write $(X_1 + \ldots + X_n)^{d-1}$ and $(X_1 + \ldots + X_n)^d$ as the following polynomials:

$$(X_1 + \ldots + X_n)^{d-1} = \sum_{\substack{\beta \in \mathbb{N}_0^n \\ ||\beta||_1 = d-1}} u_\beta \cdot X^\beta, \tag{2.104}$$

$$(X_1 + \ldots + X_n)^d = \sum_{\substack{\gamma \in \mathbb{N}_0^n \\ ||\gamma||_1 = d}} \overline{u_\gamma} \cdot X^\gamma. \tag{2.105}$$

Because of $(X_1 + \ldots + X_n)^d = (X_1 + \ldots + X_n)^{d-1} \cdot (X_1 + \ldots + X_n)$, we get

$$\overline{u_\gamma} = \sum_{\substack{i \in [n] \\ \gamma_i > 0}} u_{\gamma - e_i} \tag{2.106}$$

for $\gamma \in \mathbb{N}_0^n$ with $||\gamma||_1 = d$. From our induction hypothesis it follows

$$\overline{u_\gamma} = \sum_{\substack{i \in [n] \\ \gamma_i > 0}} u_{\gamma - e_i} \leq \sum_{\substack{i \in [n] \\ \gamma_i > 0}} (d-1)! \leq d \cdot (d-1)! = d!. \tag{2.107}$$

Hence, each coefficient of $(X_1 + \ldots + X_n)^d$ is bounded by $d!$. □

We can finish the proof of the second and third part of Theorem 54 by proving the following lemma:

Lemma 59. *Let $n, d \in \mathbb{N}$ and set $L = \binom{n+d}{d}$. The matrix $W(n,d) = d! \cdot V(n,d)^{-1}$ lies in $\mathbb{Z}^{L \times L}$ and the absolute value of each entry of $W(n,d)$ is bounded by $(d!)^3$.*

Proof. Let $k \in [L]$ and set $\beta = \alpha^{(n)}(k)$. We first prove that there is a degree-d polynomial $w_\beta \in \mathbb{Z}[X]$ s.t. we have for each $j \in [L]$

$$w_\beta(\alpha_j^{(n)}) = \begin{cases} d!, & \text{if } j = k, \\ 0, & \text{if } j = 0. \end{cases} \tag{2.108}$$

Set for $z \in [0, d]$

$$h_z(Z) := \prod_{i=0}^{z-1} (Z - i). \tag{2.109}$$

Note that h_z is a univariate polynomial of degree z, and set

$$g_\beta(X_1, \ldots, X_n) := h_{\beta_1}(X_1) \cdots h_{\beta_n}(X_n) \cdot h_{d-||\beta||_1}(d - X_1 - \ldots - X_n). \tag{2.110}$$

Now, g_β is an integer polynomial of degree d over n variables.

Let $\gamma \in \mathbb{N}_0^n$ with $||\gamma||_1 \leq d$ and $\gamma \neq \beta$. We claim that $g_\beta(\gamma) = 0$. We can distinguish two cases:

Case 1: There is an $i \in [n]$ s.t. $\gamma_i < \beta_i$. In this case, we have $h_{\beta_i}(\gamma_i) = 0$. Since $h_{\beta_i}(X_i)$ divides $g_\beta(X)$, it follows $g_\beta(\gamma) = 0$.

Case 2: For each $i \in [n]$, we have $\gamma_i \geq \beta_i$. Since $\gamma \neq \beta$, there must be one $i \in [n]$ s.t. $\gamma_i > \beta_i$. In this case, we must have $d - ||\gamma||_1 < d - ||\beta||_1$. On the other hand, note that $d - ||\gamma||_1$ must be non-negative, since $||\gamma||_1 \leq d$. It follows now $h_{d-||\beta||_1}(d - ||\gamma||_1) = 0$. Since $h_{d-||\beta||_1}(d - X_1 - \ldots - X_n)$ divides $g_\beta(X)$, it follows $g_\beta(\gamma) = 0$.

Evaluating g_β at β yields

$$g_\beta(\beta) = h_{\beta_1}(\beta_1) \cdots h_{\beta_n}(\beta_n) \cdot h_{d-||\beta||_1}(d - ||\beta||_1) = \beta_1! \cdots \beta_n! \cdot (d - ||\beta||_1)!.$$

Now, set

$$w_\beta(X) := \binom{d}{\beta_1} \cdot \binom{d - \beta_1}{\beta_2} \cdot \binom{d - \beta_1 - \beta_2}{\beta_3} \cdots \binom{d - \beta_1 - \ldots - \beta_{n-1}}{\beta_n} \cdot g_\beta(X),$$

Since $\beta = \alpha^{(n)}(k)$, Eq. (2.108) now follows.

For $W(n,d)$, note that we have

$$V(n,d) \cdot \left(\text{coeff}_d(w_{\alpha^{(n)}(1)}) \quad \ldots \quad \text{coeff}_d(w_{\alpha^{(n)}(L)})\right) \tag{2.111}$$

$$= \begin{pmatrix} w_{\alpha^{(n)}(1)}(\alpha^{(n)}(1)) & \cdots & w_{\alpha^{(n)}(L)}(\alpha^{(n)}(1)) \\ \vdots & \ddots & \vdots \\ w_{\alpha^{(n)}(1)}(\alpha^{(n)}(L)) & \cdots & w_{\alpha^{(n)}(L)}(\alpha^{(n)}(L)) \end{pmatrix} = d! \cdot \mathsf{id}_{L \times L}. \tag{2.112}$$

It follows that $W(n,d)$ is the $L \times L$-matrix whose k-th column equals the coefficient vector of $w_{\alpha^{(n)}(k)}$. Since each $w_{\alpha^{(n)}(k)}$ has integer coefficients, it follows that $W(n,d) = d! \cdot V(n,d)^{-1}$ must be an integer matrix.

Finally, we need to bound the absolute values of the entries of $W(n,d)$. For this end, it suffices to show that the absolute values of the coefficients of g_β are bounded by $(d!)^2$.

Note that g_β is the product of $h_{\beta_1}(X), \ldots, h_{\beta_n}(X), h_{d-||\beta||_1}(d - X_1 - \ldots - X_n)$. For each $i \in [n]$, h_{β_i} is the product of β_i linear functions whose coefficients are bounded by $1, 1, 2, \ldots, \beta_i - 1$. For $h_{d-||\beta||_1}(d - X_1 - \ldots - X_n)$, we have

$$\begin{aligned} & h_{d-||\beta||_1}(d - X_1 - \ldots - X_n) && (2.113) \\ = & \prod_{i=0}^{d-||\beta||_1-1} (d - X_1 - \ldots - X_n - i) && (2.114) \\ = & \prod_{j=||\beta||_1+1}^{d} (j - X_1 - \ldots - X_n). && (2.115) \end{aligned}$$

Hence, $h_{d-||\beta||_1}(d - X_1 - \ldots - X_n)$ can be written as product of linear factors whose coefficients are bounded by $||\beta||_1 + 1, \ldots, d$.

According to Lemma 58, the absolute value of each coefficient of $g_\beta(X) = h_{\beta_1}(X) \cdots h_{\beta_n}(X) \cdot h_{d-||\beta||_1}(d - X_1 - \ldots - X_n)$ is bounded by $(\beta_1 - 1)! \cdots (\beta_n - 1)! \cdot ((||\beta||_1 + 1) \cdots d) \cdot d! \leq (d!)^2$. □

We showed in Lemma 59 that each entry of $W(n,d)$ is bounded by $(d!)^3$. Now, we can bound $||W(n,d)||_\infty$ by the number of columns of $W(n,d)$ times $(d!)^3$. Hence, we have

$$||W(n,d)||_\infty \leq L \cdot (d!)^3 = \binom{n+d}{d} \cdot (d!)^3 = d! \cdot (n+d)!. \qquad (2.116)$$

Because of Lemma 56 and Proposition 57, we know that $||W(n,d)||_\infty$ is upper bounded by $||W(d,d)||_\infty \leq d! \cdot (2d)!$ for all $n, d \in \mathbb{N}$. This completes the proof of the second and third part of Theorem 54.

We will spend the rest of this subsection on calculating the determinant of $V(n,d)$. For this end, we will need to introduce several definitions:

Definition 32. Let $n, d \in \mathbb{N}$ and set $L := \binom{n+d}{d}$. We will call a list of points $x_1, \ldots, x_L \in \mathbb{Z}^n$ a **point basis** if $\det V(x_1, \ldots, x_L) \neq 0$.

Definition 33 (Translation and Transformation of Polynomials)**.** In this subsection, for $t \in \mathbb{R}^n$, we will denote by ϕ_t the linear map that is given by

$$\begin{aligned} \phi_t : \mathbb{R}[X_1, \ldots, X_n]^{\leq d} &\longrightarrow \mathbb{R}[X_1, \ldots, X_n]^{\leq d} && (2.117) \\ f(X) &\longmapsto f(X + t). && (2.118) \end{aligned}$$

By $M_t \in \mathbb{R}^{L \times L}$, we will denote the matrix representation of ϕ_t, which will depend on the currently considered monomial basis of $\mathbb{R}[X]$.

For $Q \in \mathbb{R}^{n \times n}$, we will denote by ϕ_Q the linear map that is given by

$$\begin{aligned} \phi_Q : \mathbb{R}[X_1, \ldots, X_n]^{\leq d} &\longrightarrow \mathbb{R}[X_1, \ldots, X_n]^{\leq d} && (2.119) \\ f(X) &\longmapsto f(Q \cdot X). && (2.120) \end{aligned}$$

By $M_Q \in \mathbb{R}^{L\times L}$, we will denote the matrix representation of ϕ_Q, which depends—again—on the chosen monomial basis of $\mathbb{R}[X]$.

Note that ϕ_t, M_t, ϕ_Q and M_Q depend on the degree d. By abuse of notation, we will not explicitly include d in their symbols. Typically, d can be inferred from the context.

We will show in the following that the determinant, $\det V(x_1, \ldots, x_L)$, does not change when we translate the points $x_1, \ldots, x_L$ by t or rotate them using an orthogonal matrix Q:

Lemma 60. *Let $t \in \mathbb{R}^n$. For ϕ_t from Definition 33, we have $\det \phi_t = 1$.*

Proof. We will prove that each eigenvalue of ϕ_t is one. Let $\nu \in \mathbb{C}$ be an eigenvalue of ϕ_t with eigenvector $f \in \mathbb{R}[X]^{\leq d} \setminus \{0\}$. If f is constant, then we have $\phi_t(f) = f$ and ν must be one. Otherwise, we can write f as sum $f = g + h$ s.t. g is homogenous of degree $\deg f$, and the degree of h is smaller than the degree of f. We also rewrite $\phi_t(f) = f(X + t)$ as sum $\phi_t(f) = g' + h'$ with g' homogenous and $\deg h' < \deg \phi_t(f)$. However, shifting a polynomial does not change the coefficients of its highest degree monomials. This implies that g' equals g. It follows

$$g + h' = g' + h' = \phi_t(f) = \nu \cdot f = \nu \cdot g + \nu \cdot h. \tag{2.121}$$

Rearranging terms yields $(1 - \nu)g = \nu h - h'$. Since we have $\deg g = \deg f > \deg(\nu h - h')$, the formula can only be fulfilled if $\nu = 1$. It follows that each eigenvalue of ϕ_t is one and, since the determinant is the product of eigenvalues, $\det \phi_t$ must be one, too. □

Lemma 61. *Let $Q \in \mathbb{R}^{n\times n}$. For ϕ_Q from Definition 33, we have $\det \phi_Q = \det Q^{\binom{n+d}{d-1}}$.*

Proof. Note that ϕ_Q preserves homogenous polynomials, hence, we can restrict ϕ_Q to the vector space $\mathbb{R}[X]^d$ of homogenous polynomials of degree d, which yields $\phi_d : \mathbb{R}[X]^d \to \mathbb{R}[X]^d$. For the determinant of ϕ_Q, we now have

$$\det \phi_Q = \det \phi_0 \cdots \det \phi_d = \det \phi_1 \cdots \det \phi_d. \tag{2.122}$$

It is easy to see that Q is a matrix representation of ϕ_1. Now, if $\det Q = 0$, then, we also have $\det \phi_1 = 0$ and the claim follows. Otherwise, Q is regular and can be written as the product $Q = U \cdot D \cdot R$, where U is a lower triangular matrix with ones on its diagonal, D is a diagonal matrix and R is an upper triangular matrix with ones on its diagonal. The determinant of ϕ_d is the product of the determinants of the corresponding maps induced by the matrices U, D, R. Hence, it suffices to consider triangular and diagonal matrices separately:

1. If $Q = D$ is diagonal with $\mu_1, \ldots, \mu_n$ on its diagonal, then we have for each monomial $X^\beta \in \mathbb{R}[X]^d$

   $$\phi_d(X^\beta) = \mu^\beta \cdot X^\beta. \tag{2.123}$$

   Hence, each monomial X^β is an eigenvector of ϕ_d with eigenvalue μ^β. For the determinant of ϕ_d it follows

   $$\det \phi_d = \prod_{\beta \in \mathbb{N}_0^n, ||\beta||_1 = d} \mu^\beta = \mu^{\sum_{\beta\in\mathbb{N}_0^n, ||\beta||_1 = d} \beta}. \tag{2.124}$$

Now, there are $\binom{n+d-1}{d}$ multi-indices β with $||\beta||_1 = d$. For their sum, we have $\left|\left|\sum_\beta \beta\right|\right|_1 = d \cdot \binom{n+d-1}{d}$. Since all coordinates of $\sum_\beta \beta$ must be equal, it follows that the value of each coordinate is $\frac{d}{n} \cdot \binom{n+d-1}{d} = \binom{n+d-1}{d-1}$. Hence,

$$\det \phi_d = (\mu_1 \cdots \mu_n)^{\binom{n+d-1}{d-1}} = \deg D^{\binom{n+d-1}{d-1}}. \tag{2.125}$$

2. Now, let $Q = R$ be an upper triangular matrix. Note that R can be written as a product $R = R_{n-1} \cdots R_1$ s.t. each Ri is an upper triangular matrix with ones on its diagonal and only the i-th row is a non-unit vector. Hence, we will only consider the case $R = R_i$ here, i.e., we will—without loss of generality—assume that every row, except of the i-th row of R, is a unit vector. Let

$$(0, \ldots, 0, 1, r_{i+1}, \ldots, r_n) \tag{2.126}$$

be the i-th row of R.

For a multi-index $\beta \in \mathbb{N}_0^n$, define its *literal weight* by

$$w(\beta) := \sum_{i=1}^{n} \beta_i \cdot i. \tag{2.127}$$

Set $L = \binom{n+d-1}{d}$ and let $X^{\beta_1}, \ldots, X^{\beta_L}$ be a monomial basis of $\mathbb{R}[X]^d$ s.t. the literal weights are ascending, i.e.,

$$i < j \implies w(\beta_i) \leq w(\beta_j). \tag{2.128}$$

Further, let $\gamma = (\gamma_1, \ldots, \gamma_n) \in \mathbb{N}_0^n$ be any multi-index with $||\gamma||_1 = d$. We have

$$\phi_d(X^\gamma) \tag{2.129}$$

$$=X_1^{\gamma_1} \cdots X_{i-1}^{\gamma_{i-1}} \cdot (X_i + r_{i+1}X_{i+1} + \ldots + r_n X_n)^{\gamma_i} \cdot X_{i+1}^{\gamma_{i+1}} \cdots X_n^{\gamma_n} \tag{2.130}$$

$$=X^\gamma + \sum_{\delta \in \mathbb{N}_0^n, w(\delta) > w(\gamma)} s_\delta \cdot X^\delta \tag{2.131}$$

for appropriate scalars $s_\delta \in \mathbb{R}$. I.e., $\phi_d(X^\gamma)$ can be written as X^γ plus monomials whose literal weights are higher than γ's weight. If we consider a matrix representation of ϕ_d with respect to the basis $X^{\beta_1}, \ldots, X^{\beta_L}$, then the columns of all monomials of weight higher than X^γ will appear right from the column of X^γ. It follows that the matrix representation of ϕ_d is itself an upper triangular matrix with ones on its diagonal. Hence, $\det \phi_d = 1$.

Now, it follows for any matrix $Q \in \mathbb{R}^{n \times n}$

$$\det \phi_d = \det Q^{\binom{n+d-1}{d-1}}. \tag{2.132}$$

For the determinant of the map ϕ we therefore have

$$\det \phi = \det \phi_1 \cdots \det \phi_d = \det Q^{\sum_{i=1}^{d} \binom{n+i-1}{i-1}} = \det Q^{\binom{n+d}{d-1}}. \qquad \square$$

Lemma 62. *Let $x_1, \ldots, x_L \in \mathbb{R}^n$ be a point basis. For $t \in \mathbb{R}^n$, we have*

$$V(x_1 + t, \ldots, x_L + t) = V(x_1, \ldots, x_L) \cdot M_t. \tag{2.133}$$

For any regular $Q \in \mathbb{R}^{n \times n}$, we have

$$V(Q \cdot x_1, \ldots, Q \cdot x_L) = V(x_1, \ldots, x_L) \cdot M_Q. \tag{2.134}$$

Proof. We first claim that $\det V(x_1, \ldots, x_L)$ is non-zero iff $\det V(x_1+t, \ldots, x_L+t)$ is non-zero. In fact, $x_1, \ldots, x_L$ is a point basis iff each polynomial f of degree $\leq d$ is uniquely determined by its evaluations on $x_1, \ldots, x_L$. However, the values of $f(X)$ on $x_1, \ldots, x_L$ equal the values of $f(X - t)$ on $x_1 + t, \ldots, x_L + t$. Therefore, the behaviour of each degree-d polynomial is uniquely determined by $x_1, \ldots, x_L$ iff it is uniquely determined by $x_1 + t, \ldots, x_L + t$.

Now, assume that $\det V(x_1, \ldots, x_L)$ is non-zero, and consider the $L \times L$-matrix

$$M := V(x_1 + t, \ldots, x_L + t)^{-1} \cdot V(x_1, \ldots, x_L). \tag{2.135}$$

If $c \in \mathbb{R}^L$ describes the coefficients of a polynomial f, then we have

$$V(x_1, \ldots, x_L) \cdot c = (f(x_1), \ldots, f(x_L)). \tag{2.136}$$

Further, $V(x_1+t, \ldots, x_L+t)^{-1} \cdot (f(x_1), \ldots, f(x_L))$ must describe the coefficients of a polynomial that maps each $x_i + t$ to $f(x_i)$. This polynomial is exactly $f(X - t)$. It follows that M maps the coefficients of the polynomial $f(X)$ to the coefficients of the polynomial $f(X - t)$. Hence, M is a matrix representation of ϕ_{-t}.

Analogously, for a regular matrix Q, $\det V(x_1, \ldots, x_L)$ is non-zero iff $\det V$ $(Qx_1, \ldots, Qx_L)$ is non-zero, and $V(Qx_1, \ldots, Qx_L)^{-1} \cdot V(x_1, \ldots, x_L)$ is a matrix representation of $\phi_{Q^{-1}}$. □

To prove Lemma 64, which states that $V(n, d)$ admits a nice block decomposition, we also need the following small observation:

Lemma 63. *Set $L = \binom{n+d}{d}$ and $L'' = \binom{n+d-1}{d-1}$. Let $b_1, \ldots, b_{L''}$ be an enumeration of the set*

$$\{\beta \in \mathbb{N}_0^n \mid \beta_n \neq 0, ||\beta||_1 \leq d\}. \tag{2.137}$$

Then, we have

$$\prod_{i=1}^{L''} y_i = \prod_{i=1}^{d} (d + 1 - i)^{\binom{n-1+i-1}{i-1}} \tag{2.138}$$

where $y_1, \ldots, y_{L''}$ denote the last coordinates of $b_1, \ldots, b_{L''}$.

Proof. Let $j \in [d]$ and set

$$S_j = \{\beta \in \mathbb{N}_0^n \mid \beta_n = j, ||\beta||_1 \leq d\}. \tag{2.139}$$

Since S_i is bijective to $\left\{(x_1, \ldots, x_{n-1}) \in \mathbb{N}_0^{n-1} \;\middle|\; x_1 + \ldots + x_{n-1} \leq d - j\right\}$, we have $\#S_i = \binom{n-1+d-j}{d-j}$. For the product $y_1 \cdots y_{L''}$, we have

$$\prod_{i=1}^{L''} y_i = \prod_{j=1}^{d} j^{\#S_j} = \prod_{j=1}^{d} j^{\binom{n-1+d-j}{d-j}} = \prod_{k=1}^{d} (d + 1 - k)^{\binom{n-1+k-1}{k-1}}. \qquad \square$$

Lemma 64. *Let $n, d > 2$ and $L' = \binom{n-1+d}{d}$ and $L'' = \binom{n+d-1}{d-1}$. There is a permutation matrix $P(n,d) \in \mathbb{Z}^{L \times L}$ and a diagonal matrix $D(n,d) \in \mathbb{Z}^{L'' \times L''}$ s.t.*

$$P(n,d) \cdot V(n,d) \cdot P^{-1}(n,d) = \begin{pmatrix} V(n-1,d) & 0 \\ * & D(n,d) \cdot V(n,d-1) \cdot M_{(0,\ldots,0,1)} \end{pmatrix}$$

and

$$\det D(n,d) = \prod_{i=1}^{d} (d+1-i)^{\binom{n-1+i-1}{i-1}}. \tag{2.140}$$

Proof. To prove this lemma, we introduce the following maps:

$$\begin{aligned} \pi : \mathbb{N}_0^{n-1} &\longrightarrow \mathbb{N}_0^n && (2.141)\\ (c_1, \ldots, c_{n-1}) &\longmapsto (c_1, \ldots, c_{n-1}, 0), && (2.142)\\ \tau : \mathbb{N}_0^n &\longrightarrow \mathbb{N}_0^{n-1} \times \mathbb{N} && (2.143)\\ (c_1, \ldots, c_n) &\longmapsto (c_1, \ldots, c_{n-1}, c_n + 1). && (2.144) \end{aligned}$$

π extends its input by a zero, and τ increments the last coordinate of its input. For $L' = \binom{n-1+d}{d}$ and $L'' = \binom{n+d-1}{d-1}$, we have $L = L' + L''$. We can now partition the points $\alpha^{(n)}(1), \ldots, \alpha^{(n)}(L)$ into two sets A, B, s.t. A contains all points whose last coordinate is zero and B contains all points whose last coordinate lies in $[d]$. We have

$$\begin{aligned} A :=& \{a_1, \ldots, a_{L'}\} := \left\{ \pi(\alpha^{(n-1)}(1)), \ldots, \pi(\alpha^{(n-1)}(L')) \right\} && (2.145)\\ B :=& \{b_1, \ldots, b_{L''}\} := \left\{ \tau(\alpha^{(n)}(1)), \ldots, \tau(\alpha^{(n)}(L'')) \right\}. && (2.146) \end{aligned}$$

Indeed, if the last coordinate of $\alpha^{(n)}(i)$ is zero, then $\alpha^{(n)}(i)$ is of shape $\pi(\alpha^{(n-1)}(j))$. Otherwise, we know that $\alpha^{(n)}(i) - (0, \ldots, 0, 1)$ lies in $\mathbb{N}_0^n$ and has a one-norm of $\leq d - 1$, hence, it must be of shape $\alpha^{(n)}(j)$ for some $j < L''$.

Now, we want to rearrange the columns and rows of $V(n,d)$. Note, that each row of $V(n,d)$ corresponds to a point $\alpha^{(n)}(i)$ and that each column of $V(n,d)$ corresponds to a multi-index $\alpha^{(n)}(j)$. We arrange the columns and rows of $V(n,d)$ s.t. all columns and rows that correspond to points in A are on the left resp. on the top, while all columns and rows that correspond to points in B are on the right resp. on the bottom. Note, that whenever we swap the i-th and j-th column of $V(n,d)$, we also swap the i-th and j-th row of $V(n,d)$. Hence, there is a permutation matrix $P(n,d)$ s.t.

$$P(n,d) \cdot V(n,d) \cdot P^{-1}(n,d) = \begin{pmatrix} a_1^{a_1} & \ldots & a_1^{a_{L'}} & a_1^{b_1} & \ldots & a_1^{b_{L''}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{L'}^{a_1} & \ldots & a_{L'}^{a_{L'}} & a_{L'}^{b_1} & \ldots & a_{L'}^{b_{L''}} \\ b_1^{a_1} & \ldots & b_1^{a_{L'}} & b_1^{b_1} & \ldots & b_1^{b_{L''}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ b_{L''}^{a_1} & \ldots & b_{L''}^{L'} & b_{L''}^{b_1} & \ldots & b_{L''}^{b_{L''}} \end{pmatrix}. \tag{2.147}$$

Set $C := P(n,d) \cdot V(n,d) \cdot P^{-1}(n,d)$, and, for $i \in [L'']$, denote by y_i the last coordinate of b_i. Note that y_i must be a number between one and d. For $i,j \in [L']$, the (i,j)-th entry of C is given by

$$a_i^{a_j} = \pi(\alpha^{(n-1)}(i))^{\pi(\alpha^{(n-1)}(j))} = \alpha^{(n-1)}(i)^{\alpha^{(n-1)}(j)}. \tag{2.148}$$

It follows that the top left $L' \times L'$-submatrix of C is $V(n-1,d)$. If $i \in [L']$ and $j \in [L'']$, we have for the $(i, L'+j)$-th entry of C

$$a_i^{b_j} = \pi(\alpha^{(n-1)}(i))^{\tau(\alpha^{(n)}(j))} = 0, \tag{2.149}$$

since the last coordinate of a_i is zero, while the last coordinate of b_j is at least one. Hence, the top right $L' \times L''$-submatrix of C must be zero. For $i,j \in [L'']$, we have for $(L'+i, L'+j)$-th entry of C

$$b_i^{b_j} = \tau(\alpha^{(n)}(i))^{\tau(\alpha^{(n)}(j))} = (\alpha^{(n)}(i) + (0,\dots,0,1))^{\alpha^{(n)}(j)+(0,\dots,0,1)} \tag{2.150}$$

$$= y_i \cdot (\alpha^{(n)}(i) + (0,\dots,0,1))^{\alpha^{(n)}(j)}. \tag{2.151}$$

Denote by $D(n,d) = \operatorname{diag}(y_1,\dots,y_{L''})$ the diagonal matrix with $y_1,\dots,y_{L''}$ on its diagonal. Then, the bottom right $L'' \times L''$-submatrix of C equals

$$D(n,d) \cdot V(\alpha^{(n)}(1) + (0,\dots,0,1), \dots, \alpha^{(n)}(L'') + (0,\dots,0,1)). \tag{2.152}$$

Because of Lemma 63, we know that the determinant of $D(n,d)$ equals $\prod_{i=1}^{d}(d+1-i)^{\binom{n-1+i-1}{i-1}}$.

Finally, $V(\alpha^{(n)}(1) + (0,\dots,0,1), \dots, \alpha^{(n)}(L'') + (0,\dots,0,1))$ is a Vandermonde matrix whose interpolation points have been shifted by $(0,\dots,0,1)$. Because of Lemma 62, we know that

$$V(\alpha^{(n)}(1) + (0,\dots,0,1), \dots, \alpha^{(n)}(L'') + (0,\dots,0,1)) \tag{2.153}$$

$$= V(\alpha^{(n)}(1), \dots, \alpha^{(n)}(L'')) \cdot M_{(0,\dots,0,1)}. \tag{2.154}$$

This finishes the proof. □

To complete the proof of Theorem 54, we will first explicitly compute the determinant of the Vandermonde matrix in the cases $d = 1$ and $n = 1$. The general case then follows inductively by using Lemma 64.

Lemma 65. *Let $n \in \mathbb{N}$. For $x_1,\dots,x_{n+1} \in \mathbb{R}^n$, we have*

$$\det V(x_1,\dots,x_{n+1}) = \det \begin{pmatrix} x_2 - x_1 \\ \vdots \\ x_{n+1} - x_1 \end{pmatrix}. \tag{2.155}$$

In particular, we have $\det V(n,1) = 1$.

Proof. Note that $V(x_1,\dots,x_{n+1})$ is given by

$$V(x_1,\dots,x_{n+1}) = \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_{n+1} \end{pmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}. \tag{2.156}$$

Since the determinant of a matrix does not change when we subtract rows of the matrix from other rows, the determinant of $V(x_1, \dots, x_{n+1})$ is equal to

$$\det \begin{pmatrix} 1 & x_1 \\ 0 & x_2 - x_1 \\ \vdots & \vdots \\ 0 & x_{n+1} - x_1 \end{pmatrix} = \det \begin{pmatrix} x_2 - x_1 \\ \vdots \\ x_{n+1} - x_1 \end{pmatrix}. \tag{2.157}$$

For $V(n, 1)$, note that $\alpha^{(1)}(1), \dots, \alpha^{(1)}(n+1)$ enumerate the elements of the set

$$\left\{ \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix}, \dots, \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix} \right\}. \tag{2.158}$$

By reordering $\alpha^{(1)}(1), \dots, \alpha^{(1)}(n+1)$, we have for an appropriate permutation matrix $P \in \mathbb{Z}^{(n+1)\times(n+1)}$

$$P \cdot V(n, 1) \cdot P^{-1} = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 1 & 1 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 1 & 0 & \dots & 1 \end{pmatrix}. \tag{2.159}$$

Hence, we have $\det V(n, 1) = 1$. □

Lemma 66. *For $d \in \mathbb{N}$ and $x_1, \dots, x_{d+1} \in \mathbb{R}$, we have*

$$\det V(x_1, \dots, x_{d+1}) = \prod_{1 \le i < j \le d} (x_j - x_i). \tag{2.160}$$

In particular, we have

$$\det V(1, d) = d! \cdot (d-1)! \cdots 2! \cdot 1! = \prod_{i=1}^{d} (d+1-i)^i = \prod_{i=1}^{d} (d+1-i)^{i \cdot \binom{1-1+i}{i}}.$$

Proof. Because of Lemmas 60 and 62, the determinant of $V(x_1, \dots, x_{d+1})$ equals the determinant of $V(0, x_2 - x_1, \dots, x_d - x_1)$, which admits the following block decomposition:

$$V(0, x_2 - x_1, \dots, x_d - x_1) \tag{2.161}$$

$$= \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & x_2 - x_1 & (x_2 - x_1)^2 & \dots & (x_2 - x_1)^d \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{d+1} - x_1 & (x_{d+1} - x_1)^2 & \dots & (x_{d+1} - x_1)^d \end{pmatrix} \tag{2.162}$$

$$= \begin{pmatrix} 1 & 0 \\ * & D \cdot V(x_2 - x_1, \dots, x_{d+1} - x_1) \end{pmatrix}, \tag{2.163}$$

where D is a diagonal matrix with the values $x_2 - x_1, \dots, x_{d+1} - x_1$ on its diagonal. We now have.

$$\begin{aligned}
\det V(x_1, \dots, x_{d+1}) &= \det V(0, x_2 - x_1, \dots, x_d - x_1) && (2.164)\\
&= \det D \cdot \det V(x_2 - x_1, \dots, x_{d+1} - x_1) && (2.165)\\
&= \prod_{i=2}^{d+1} (x_i - x_1) \cdot \det V(x_2 - x_1, \dots, x_{d+1} - x_1). && (2.166)
\end{aligned}$$

By Lemmas 60 and 62, it follows $\det V(x_2 - x_1, \dots, x_{d+1} - x_1) = \det V(x_2, \dots, x_{d+1})$. By an inductive argument on d, we have

$$\det V(x_2, \dots, x_{d+1}) = \prod_{2 \leq i < j \leq d} (x_j - x_i). \qquad (2.167)$$

Hence, we have

$$\begin{aligned}
\det V(x_1, \dots, x_{d+1}) &= \prod_{i=2}^{d+1} (x_i - x_1) \cdot \det V(x_2 - x_1, \dots, x_{d+1} - x_1) && (2.168)\\
&= \prod_{i=2}^{d+1} (x_i - x_1) \cdot \prod_{2 \leq i < j \leq d} (x_j - x_i) && (2.169)\\
&= \prod_{1 \leq i < j \leq d} (x_j - x_i). && (2.170)
\end{aligned}$$

For $x_1 = 0, \dots, x_{d+1} = d$, this product is equal to $d! \cdot (d-1)! \cdots 2! \cdot 1!$. Hence, we have

$$\begin{aligned}
\det V(1, d) =& d! \cdot (d-1)! \cdots 2! \cdot 1! = \prod_{i=1}^{d} (d + 1 - i)^i && (2.171)\\
=& \prod_{i=1}^{d} (d + 1 - i)^{i \cdot \binom{1-1+i}{i}}. && \square
\end{aligned}$$

We will now finish the proof of Theorem 54:

Proof Theorem 54, Part 1. By Lemmas 65 and 66 the formula

$$\det V(n, d) = \prod_{i=1}^{d} (d + 1 - i)^{i \cdot \binom{n-1+i}{i}} \qquad (2.172)$$

is already proven in case of $n = 1$ or $d = 1$. Hence, let $n > 1$ and $d > 1$. We will prove Eq. (2.172) by a structural induction on the pair (n, d), i.e., we assume the formula is proven for $V(n-1, d)$ and $V(n, d-1)$. Because of Lemma 64, we have

$$\begin{aligned}
&\det V(n, d) && (2.173)\\
&= \det V(n-1, d) \cdot \det(V(n, d-1)) \cdot \det D(n, d) \cdot \det M_{(0,\dots,0,1)} && (2.174)\\
&= \det V(n-1, d) \cdot \det(V(n, d-1)) \cdot \prod_{i=1}^{d} (d + 1 - i)^{\binom{n-1+i-1}{i-1}} && (2.175)
\end{aligned}$$

where the last equality follows from Lemma 60 and Lemma 63. By using our induction hypothesis, we can replace $\det V(n-1,d)$ by $\prod_{i=1}^{d}(d+1-i)^{i\cdot\binom{n-2+i}{i}}$ and $\det V(n,d-1)$ by $\prod_{i=1}^{d-1}(d-i)^{i\cdot\binom{n-1+i}{i}} = \prod_{i=1}^{d}(d+1-i)^{(i-1)\cdot\binom{n-1+i-1}{i-1}}$. It follows

$$\det V(n,d) \tag{2.176}$$

$$= \det V(n-1,d) \cdot \det V(n,d-1) \cdot \left(\prod_{i=1}^{d}(d+1-i)^{\binom{n-1+i-1}{i-1}}\right) \tag{2.177}$$

$$= \left(\prod_{i=1}^{d}(d+1-i)^{i\cdot\binom{n-2+i}{i}}\right) \cdot \left(\prod_{i=1}^{d}(d+1-i)^{(i-1)\cdot\binom{n-1+i-1}{i-1}}\right) \tag{2.178}$$

$$\cdot \left(\prod_{i=1}^{d}(d+1-i)^{\binom{n-1+i-1}{i-1}}\right) \tag{2.179}$$

$$= \prod_{i=1}^{d}(d+1-i)^{i\cdot\binom{n-2+i}{i}+(i-1)\cdot\binom{n-1+i-1}{i-1}+\binom{n-1+i-1}{i-1}} \tag{2.180}$$

$$= \prod_{i=1}^{d}(d+1-i)^{i\cdot\binom{n-2+i}{i}+i\cdot\binom{n-1+i-1}{i-1}} \tag{2.181}$$

$$= \prod_{i=1}^{d}(d+1-i)^{i\cdot\binom{n-1+i}{i}}. \tag{2.182}$$

Hence, Eq. (2.172) does hold for all $n, d \in \mathbb{N}$. □

2.2.2 Limits on Our Bounds

In this subsection, we will discuss the quality of the quasi-inverse $W(n,d)$ and the tightness of the bound we give for the interpolation number Γ_d.

On the Optimality of $W(n,d)$. $W(n,d)$ is the best quasi-inverse of $V(n,d)$ we can achieve in the following sense: for each $c \in \mathbb{N}$ and $W' \in \mathbb{Z}^{L\times L}$ satisfying

$$W' \cdot V(n,d) = c \cdot \mathsf{id}_{L\times L}, \tag{2.183}$$

$d!$ must divide c.

This is, in fact, easy to show: note that the first column of the inverse $V(1,d)^{-1}$ is the coefficient vector of the polynomial

$$f(X) := \frac{1}{d!}\prod_{i=1}^{d}(i-X). \tag{2.184}$$

This is, because f is the unique degree-d polynomial with $f(0) = 1$ and $f(1) = \ldots = f(d) = 0$. Now, the leading term of f is $\pm\frac{1}{d!}$. Hence, the matrix $V(1,d)^{-1}$ contains the entry $\pm\frac{1}{d!}$. Because of Lemma 64, $V(n,d)^{-1}$ contains $V(1,d)^{-1}$ as submatrix. Therefore, $V(n,d)^{-1}$ also contains the entry $\frac{1}{d!}$. Hence, whenever we have $c \cdot V(n,d)^{-1} \in \mathbb{Z}^{L\times L}$, it must follow $d!|c$. Ergo, $W(n,d)$ is optimal, i.e., it is the smallest integer matrix that multiplied with $V(n,d)$ yields a positive multiple of the identity.

On the Bound of Γ_d. For $d \in \mathbb{N}$, we could bound the interpolation number Γ_d from above by $(2d)!$. This bound is not tight. In fact, Table 2.1 lists Γ_d, for $d = 1, \ldots, 8$, and the ratio $\frac{(2d)!}{\Gamma_d}$ of our bound to the interpolation number.

From Table 2.1, we can deduce two things: first, Γ_d seems to grow exponentially in d. Second, our bound of $(2d)!$ is evidently not tight as the entries in the last column of Table 2.1 seem to grow exponentially. It might be that our bound is overly conservative and $2 \cdot d!$, for example, is a closer upper bound for Γ_d. Unfortunately, to solidify any conjecture in this regard, one would need to compute Γ_d for higher values of d. This becomes computationally intractable, since the dimensions of the involved matrices grow exponentially in d. Computing Γ_{10} already makes it necessary to invert a matrix of shape 48620×48620.

d	Γ_d	$(2d)!/\Gamma_d$
1	2	1
2	4	6
3	12	60
4	32	1260
5	88	$41236 + 4/11$
6	$261 + 1/3$	$1832914 + 2/7$
7	$725 + 1/3$	$120190658 + 14/17$
8	$2218 + 2/3$	9430344000

Table 2.1: This table lists the interpolation numbers Γ_d for $d = 1, \ldots, 8$. Additionally, in the third column the ratio $\frac{(2d)!}{\Gamma_d}$ between our bound and the interpolation number is calculated. This ration indicates the tightness of our bound: the smaller this ratio is the closer our bound comes to the genuine value of $\Gamma_d = \left|\left|V(d,d)^{-1}\right|\right|_\infty$.

2.2.3 On Random Polynomials

We will prove here Theorem 55. For this end, we will need the following simple lemma.

Lemma 67. *Let X, Y be two random variables over $\mathbb{Z}$, which do not need to be independent. If Y is bounded by $B > 0$, we have*

$$|\mathbb{E}[X \cdot Y]| \leq B \cdot \mathbb{E}[|X|] \leq B \cdot \mathbb{E}[X^2]. \tag{2.185}$$

Proof. For the first inequality, we have

$$|\mathbb{E}[X \cdot Y]| = \sum_{x,y \in \mathbb{Z}} \Pr[X = x, Y = y] \cdot x \cdot y \tag{2.186}$$

$$\leq \sum_{x,y \in \mathbb{Z}} \Pr[X = x, Y = y] \cdot |x| \cdot |y| \tag{2.187}$$

$$\leq \sum_{x,y \in \mathbb{Z}} \Pr[X = x, Y = y] \cdot |x| \cdot B \tag{2.188}$$

$$\leq B \cdot \sum_{x,y \in \mathbb{Z}} \Pr[X = x, Y = y] \cdot |x| = B \cdot \mathbb{E}[|X|]. \tag{2.189}$$

For the second inequality, we have

$$\begin{aligned}\mathbb{E}[|X|] &= \sum_{x\in\mathbb{Z}} |x| \cdot \Pr[X = x] &(2.190)\\ &\leq \sum_{x\in\mathbb{Z}} x^2 \cdot \Pr[X = x] = \mathbb{E}[X^2]. &\square\end{aligned}$$

From Lemma 67 and Theorem 54, we can now deduce the following theorem, which is more general than Theorem 55:

Theorem 68. *Let $\mathcal{D}$ be a distribution with support in $\mathbb{Z}[X]^{\leq d}$. For a random variable $f \leftarrow \mathcal{D}$, denote by $c_0, \ldots, c_d$ the random coefficients of f, i.e.,*

$$f(X) = \sum_{i=0}^{d} c_i X^i. \tag{2.191}$$

Let $B > 0$ s.t. we have for each $f \leftarrow \mathcal{D}$ and $x \in \{0, \ldots, d\}$

$$|f(x)| \leq B. \tag{2.192}$$

Further, let $e > 0$ s.t. we have for each $x \in [2d]$

$$\left| \mathop{\mathbb{E}}_{f\leftarrow\mathcal{D}} [f(x)^2 - f(0)^2] \right| \leq e. \tag{2.193}$$

For $i \in [d]$, we have

$$\mathbb{E}[c_i^2] \leq \Gamma_{2d} \cdot e \cdot (1 + 2B \cdot \Gamma_d)^{d-i}. \tag{2.194}$$

Proof. Denote by $c_0, \ldots, c_d$ the random coefficients of $f \leftarrow \mathcal{D}$. For $i > d$, we will set $c_i := 0$.

Since we have $|f(x)| \leq B$ for each $f \leftarrow \mathcal{D}$ and $x \in \{0, \ldots, d\}$, it follows from Theorem 54

$$||\mathrm{coeff}_d(f)||_\infty \leq \Gamma_d \cdot B. \tag{2.195}$$

This means that each c_i is bounded by $\Gamma_d \cdot B$.

Now, for $g(X) := \mathbb{E}[f(X)^2 - f(0)^2] \in \mathbb{R}[X]$, we have

$$g(X) = \mathbb{E}\left[\sum_{i=0}^{2d} \left(\sum_{j=0}^{i} c_j \cdot c_{i-j} \right) X^i - c_0^2 \right] = \sum_{i=1}^{2d} \left(\sum_{j=0}^{i} \mathbb{E}[c_j \cdot c_{i-j}] \right) X^i. \tag{2.196}$$

Since the values of $g(0), \ldots, g(2d)$ are bounded by e, we get

$$\left| \sum_{j=0}^{i} \mathbb{E}[c_j \cdot c_{i-j}] \right| \leq \Gamma_{2d} \cdot e \tag{2.197}$$

for each $i \in [2d]$. Now, let $k \in [d]$. We have with $i = 2k$

$$\left| \sum_{j=0}^{2k} \mathbb{E}[c_j \cdot c_{2k-j}] \right| = \left| \mathbb{E}[c_k^2] + 2 \cdot \sum_{j=k+1}^{2k} \mathbb{E}[c_j \cdot c_{2k-j}] \right| \leq \Gamma_{2d} \cdot e. \tag{2.198}$$

By the triangle inequality, we have

$$\mathbb{E}[c_k^2] \leq \Gamma_{2d} \cdot e + 2 \cdot \sum_{j=k+1}^{2k} |\mathbb{E}[c_j \cdot c_{2k-j}]|. \tag{2.199}$$

We can apply Lemma 67 on the summands of the right-hand side, where we use that c_{2k-j} is bounded by $\Gamma_d \cdot B$. This yields for each $k \in [d]$

$$\begin{aligned}\mathbb{E}[c_k^2] \leq & \Gamma_{2d} \cdot e + 2 \cdot \sum_{j=k+1}^{2k} \left(\Gamma_d \cdot B \cdot \mathbb{E}[c_j^2]\right) & (2.200)\\ = & \Gamma_{2d} \cdot e + 2B \cdot \Gamma_d \cdot \sum_{j=k+1}^{2k} \mathbb{E}[c_j^2] & (2.201)\\ \leq & \Gamma_{2d} \cdot e + 2B \cdot \Gamma_d \cdot \sum_{j=k+1}^{d} \mathbb{E}[c_j^2] & (2.202)\end{aligned}$$

We claim that we have

$$\mathbb{E}[c_k^2] \leq \Gamma_{2d} \cdot e \cdot (1 + 2B \cdot \Gamma_d)^{d-k} \tag{2.203}$$

We will prove this claim by reverse induction on $k = d, \ldots, 1$. For $k = d$, the claim follows from Eq. (2.197).
Now, let $k < d$ and assume that the bound has been proven for $\mathbb{E}[c_{k+1}^2], \ldots, \mathbb{E}[c_d^2]$. We have

$$\begin{aligned}\mathbb{E}[c_k^2] \leq & \Gamma_{2d} \cdot e + 2B \cdot \Gamma_d \cdot \sum_{j=k+1}^{d} \mathbb{E}[c_j^2] & (2.204)\\ \leq & \Gamma_{2d} \cdot e + 2B \cdot \Gamma_d \cdot \sum_{j=k+1}^{d} \Gamma_{2d} \cdot e \cdot (1 + 2B \cdot \Gamma_d)^{d-j} & (2.205)\\ = & \Gamma_{2d} \cdot e \cdot \left(1 + 2B \cdot \Gamma_d \cdot \sum_{j=k+1}^{d} (1 + 2B \cdot \Gamma_d)^{d-j}\right) & (2.206)\\ = & \Gamma_{2d} \cdot e \cdot \left(1 + 2B \cdot \Gamma_d \cdot \sum_{j=0}^{d-k-1} (1 + 2B \cdot \Gamma_d)^{j}\right) & (2.207)\\ = & \Gamma_{2d} \cdot e \cdot \left(1 + 2B \cdot \Gamma_d \cdot \frac{(1 + 2B \cdot \Gamma_d)^{d-k} - 1}{2B \cdot \Gamma_d}\right) & (2.208)\\ = & \Gamma_{2d} \cdot e \cdot (1 + 2B \cdot \Gamma_d)^{d-k}. & \square\end{aligned}$$

Observe now that we have for an integer random variable c

$$\Pr[c \neq 0] = \sum_{z \in \mathbb{Z} \setminus \{0\}} \Pr[c = z] \leq \sum_{z \in \mathbb{Z} \setminus \{0\}} \Pr[c = z] \cdot z^2 = \mathbb{E}[c^2]. \tag{2.209}$$

Hence, Theorem 68 states that the probability for being non-zero of each non-absolute coefficient $c_1, \ldots, c_d$ of $f \leftarrow \mathcal{D}$ is bounded by

$$\Pr[c_i \neq 0] \leq \mathbb{E}[c_i^2] \leq \Gamma_{2d} \cdot e \cdot (1 + 2B \cdot \Gamma_d)^{d-i}. \tag{2.210}$$

For $k \in [d]$, the degree of f is at most $d-k$ iff $c_d, \ldots, c_{d-k+1}$ are all zero. Hence, we have for $f \leftarrow \mathcal{D}$

$$\begin{aligned}
\Pr[\deg f > d-k] =& \Pr[c_d \neq 0 \vee \ldots \vee c_{d-k+1} \neq 0] && (2.211)\\
\leq& \sum_{i=d-k+1}^{d} \Pr[c_i \neq 0] && (2.212)\\
\leq& \sum_{i=d-k+1}^{d} \left(\Gamma_{2d} \cdot e \cdot (1 + 2B \cdot \Gamma_d)^{d-i}\right) && (2.213)\\
\leq& \Gamma_{2d} \cdot e \cdot \sum_{i=d-k+1}^{d} (1 + 2B \cdot \Gamma_d)^{d-i} && (2.214)\\
\leq& \Gamma_{2d} \cdot e \cdot \sum_{i=0}^{k-1} (1 + 2B \cdot \Gamma_d)^{i}. && (2.215)
\end{aligned}$$

Now, $\sum_{i=0}^{k-1}(1+2B\cdot\Gamma_d)^i$ is upper-bounded by $(1+2B\cdot\Gamma_d)^k$, since $1+2B\cdot\Gamma_d \geq 2$. This is, because we have $\Gamma_d \geq 1$ and demanded $B \geq \frac{1}{2}$ in Theorem 55. This completes the proof of Theorem 55.

2.3 On Mean Square Distinguishers

In the following, we will study the task of distinguishing different distributions. Let $\mathcal{D}_0, \mathcal{D}_1$ be two discrete and memoryless distributions over $\mathbb{R}$. Let us assume that we receive a list of N independent samples $\alpha_1, \ldots, \alpha_N$ from $\mathcal{D}_0$, a list of N independent samples $\beta_1, \ldots, \beta_N$ from $\mathcal{D}_1$ and a list of N independent samples $\gamma_1, \ldots, \gamma_N$ from $\mathcal{D}_b$ for an unknown $b \in \{0, 1\}$. Our task is to predict b, i.e., to decide if the γ-values have been sampled from $\mathcal{D}_0$ or $\mathcal{D}_1$.

We will present here a simple algorithm for this task: the *mean square distinguisher.* It works by computing the squares $\alpha_1^2, \ldots, \beta_1^2, \ldots, \gamma_1^2, \ldots$ of the received samples and then calculating their means $\overline{\alpha} := \frac{1}{N}\sum_{i=1}^{N} \alpha_i^2$, $\overline{\beta} := \frac{1}{N}\sum_{i=1}^{N} \beta_i^2$ and $\overline{\gamma} := \frac{1}{N}\sum_{i=1}^{N} \gamma_i^2$. It will then output 0, if $\overline{\gamma}$ is close enough to $\overline{\alpha}$, and 1, if $\overline{\gamma}$ is close enough to $\overline{\alpha}$.

While this is a very simple distinguishing algorithm that can be easily tricked, we will show here that this algorithm has a high distinguishing advantage if the distributions $\mathcal{D}_0$ and $\mathcal{D}_1$ are derived from random polynomials over $\mathbb{R}$ of constant degree d.

Let us first formally introduce the mean square distinguisher:

Algorithm 9 (Mean Square Distinguisher)**.** For $N \in \mathbb{N}$, the algorithm receives as input three lists of real numbers: $(\alpha_i)_{i=1}^N$, $(\beta_i)_{i=1}^N$, $(\gamma_{i=1})^N$. Additionally, it receives a parameter $r > 0$.

It assumes that $\alpha_1, \ldots, \alpha_N$ and have been sampled independently at random from a memoryless distribution $\mathcal{D}_0$ and $\mathcal{D}_1$, respectively. It tries to decide if $\gamma_1, \ldots, \gamma_N$ have been sampled from $\mathcal{D}_0$ or $\mathcal{D}_1$.

Concretely, the algorithm works as follows:

```
1: set $\overline{\alpha} := \frac{1}{N}\sum_{i=1}^{N}\alpha_i^2$
2: set $\overline{\beta} := \frac{1}{N}\sum_{i=1}^{N}\beta_i^2$
3: set $\overline{\gamma} := \frac{1}{N}\sum_{i=1}^{N}\gamma_i^2$
4: set $e_0 := |\overline{\alpha} - \overline{\gamma}|$
5: set $e_1 := |\overline{\beta} - \overline{\gamma}|$
6: if $e_0 > r > e_1$ then
7:     return 1
8: else if $e_1 > r > e_0$ then
9:     return 0
10: else
11:     draw $b' \leftarrow \{0,1\}$
12:     return $b'$
13: end if
```

Definition 34. Let $N \in \mathbb{N}$ and $r \in \mathbb{R}_{>0}$. If $\mathcal{T}$ denotes Algorithm 9 and $\mathcal{D}_0, \mathcal{D}_1$ are two memoryless distributions over $\mathbb{R}$, we define the **distinguishing advantage** of $\mathcal{T}$ by

$$\mathsf{adv}^{\mathsf{Dist}}_{\mathcal{D}_0,\mathcal{D}_1,N,r}(\mathcal{T}) := 2 \cdot \Pr[\mathcal{T}((\alpha_i)_i, (\beta_i)_i, (\gamma_i)_i, r) = b] - 1 \tag{2.216}$$

where the probability is taken over the randomness of $b \leftarrow \{0,1\}$, $\alpha_1, \dots, \alpha_N \leftarrow \mathcal{D}_0$, $\beta_1, \dots, \beta_N \leftarrow \mathcal{D}_1$ and $\gamma_1, \dots, \gamma_N \leftarrow \mathcal{D}_b$.

Note that $\mathcal{T}$ is controlled by two parameters: N and r. N determines the sample size used by $\mathcal{T}$. Intuitively, $\mathcal{T}$'s advantage should rise with N, and, in fact, we will show this in Lemma 73. On the other hand, r is a control parameter that shall ensure that $\mathcal{T}$'s success probability is close to 50%, even if we cannot guarantee a high advantage for $\mathcal{T}$. Let us explain this as follows: imagine that the if-check of Line 6 would accept iff $e_0 > e_1$ and that the if-check of Line 8 would accept iff $e_1 > e_0$. Now, consider the distributions $\mathcal{D}_0$ that outputs constantly 0, and $\mathcal{D}_1$ that outputs 0 with probability 90% and 1 with probability 10%. Given the simplicity of both distributions, we would expect $\mathcal{T}$ (without checks for r) to have a high advantage at distinguishing $\mathcal{D}_0$ from $\mathcal{D}_1$. Unfortunately, for $N = 1$, this is not true: If γ_1 is sampled from $\mathcal{D}_0$, then $\mathcal{T}$ will output 0 with a probability of 55%. On the other hand, if γ_1 is sampled from $\mathcal{D}_1$, then it can be shown that $\mathcal{T}$ will output 0 with a probability of 54%. Paradoxically, it seems that $\mathcal{T}$ is always biased towards the constant distribution. The control parameter r rectifies this behaviour, as we will show, too.

Lemma 69. *Algorithm 9 makes $\Theta(N)$ arithmetic operations over $\mathbb{R}$.*

When $\mathcal{D}_0, \mathcal{D}_1$ are B-bounded distributions over $\mathbb{Z}$, and when r can be written as $r = \frac{a}{b}$ with $a, b \in \mathbb{Z}$, $|a|, |b| \leq B$, then Algorithm 9 can be implemented by using $O(\log(B)^2 \cdot \log(N) \cdot N)$ bit operations.

Proof. It is easy to count the operations over $\mathbb{R}$ performed by Algorithm 9.

Denote Algorithm 9 by $\mathcal{T}$. If each value $\alpha_1, \dots, \alpha_N, \beta_1, \dots, \beta_N, \gamma_1, \dots, \gamma_N$ lies in $\{-B, \dots, B\}$ for $B \in \mathbb{N}$, we can slightly modify the behaviour of $\mathcal{T}$ to avoid divisions. Concretely, we calculate the sums $\overline{\alpha} = \sum_{i=1}^{N}\alpha_i^2$, $\overline{\alpha} = \sum_{i=1}^{N}\beta_i^2$ and $\overline{\alpha} = \sum_{i=1}^{N}\gamma_i^2$ in Lines 1 to 3. Calculating such a sum has a bit complexity of $O(\log(B)^2 \cdot \log(N) \cdot N)$.

In Lines 6 and 8, we then check

$$b \cdot |\overline{\alpha} - \overline{\gamma}| > N \cdot r > b \cdot |\overline{\beta} - \overline{\gamma}| \tag{2.217}$$

and

$$b \cdot |\overline{\beta} - \overline{\gamma}| > N \cdot r > b \cdot |\overline{\alpha} - \overline{\gamma}|, \tag{2.218}$$

respectively. Since the involved numbers are bounded by $B^3 \cdot N$, the bit complexity of those checks lie in $O(\log(B) \cdot \log(N) \cdot N)$. □

To prove Lemma 73, we will make use of Hoeffding's inequality for approximating means of distributions:

Theorem 70 (Hoeffding's Inequality [Hoe63])**.** *Let $N \in \mathbb{N}$ and $B, t \geq 0$. Let $x_1, \ldots, x_N$ be independent random variables with $|x_N|, \ldots, |x_N| \leq B$. It holds*

$$\Pr\left[\left|\frac{x_1 + \ldots + x_N}{B \cdot N} - \frac{\mathbb{E}[x_1] + \ldots + \mathbb{E}[x_N]}{B \cdot N}\right| \geq 2t\right] \leq 2 \cdot \exp(-2Nt^2). \quad (2.219)$$

For $N = N'^3$ and $t = \frac{1}{N'}$, we get

$$\Pr\left[\left|\frac{x_1 + \ldots + x_N}{N} - \frac{\mathbb{E}[x_1] + \ldots + \mathbb{E}[x_N]}{N}\right| \geq 2\frac{B}{N'}\right] \leq 2 \cdot \exp(-2N'). \quad (2.220)$$

Lemma 71. *Let $\mathcal{D}_0$ be a memoryless distributions over $\mathbb{R}$ that is bounded by some $B > 0$, i.e., its support lies in $[-B, B]$.*

For $N' \in \mathbb{N}$, set $N = N'^3$. Draw

$$\alpha_1, \ldots, \alpha_N, \gamma_1, \ldots, \gamma_N \leftarrow \mathcal{D}_0 \quad (2.221)$$

and set $\overline{\alpha} = \frac{1}{N}\sum_{i=1}^N \alpha_i^2$ and $\overline{\gamma} = \frac{1}{N}\sum_{i=1}^N \gamma_i^2$. For $r \geq 4B^2/N'$, we have

$$\Pr[|\overline{\alpha} - \overline{\gamma}| > r] < 4\exp(-2N'). \quad (2.222)$$

Proof. Set $\mu_0 := \mathbb{E}_{x \leftarrow \mathcal{D}_0}[x^2]$. According to Theorem 70, we have

$$\Pr\left[|\overline{\alpha} - \mu_0| \geq 2\frac{B^2}{N'}\right] \leq= 2\exp(-2N'), \quad (2.223)$$

$$\Pr\left[|\overline{\gamma} - \mu_0| \geq 2\frac{B^2}{N'}\right] \leq= 2\exp(-2N'). \quad (2.224)$$

By a union bound, both inequalities $|\overline{\alpha} - \mu_0| < 2\frac{B^2}{N'}$ and $|\overline{\gamma} - \mu_0| < 2\frac{B^2}{N'}$ will hold with probability $> 1 - 4\exp(-2N')$. Hence, by triangle inequality, the inequality

$$|\overline{\alpha} - \overline{\gamma}| < 4B^2/N' \quad (2.225)$$

must hold with probability $> 1 - 4\exp(-2N')$. □

Lemma 72. *Let $\mathcal{D}_0$, $\mathcal{D}_1$ be two memoryless B-bounded distributions over $\mathbb{R}$. Set*

$$e := \left|\mathop{\mathbb{E}}_{x \leftarrow \mathcal{D}_0}[x^2] - \mathop{\mathbb{E}}_{y \leftarrow \mathcal{D}_1}[y^2]\right|. \quad (2.226)$$

Let $N' \in \mathbb{N}$ and set $N = N'^3$. Draw

$$\beta_1, \ldots, \beta_N \leftarrow \mathcal{D}_1 \qquad \textit{and} \qquad \gamma_1, \ldots, \gamma_N \leftarrow \mathcal{D}_0. \quad (2.227)$$

For $\overline{\beta} = \sum_{i=1}^N \beta_i^2$ and $\overline{\gamma} = \sum_{i=1}^N \gamma_i^2$, we have

$$\Pr\left[|\overline{\beta} - \overline{\gamma}| < e/2\right] < 4\exp\left(-N \cdot e^2/(32 \cdot B^4)\right) \quad (2.228)$$

Proof. Set $\mu_0 := \mathbb{E}_{x \leftarrow \mathcal{D}_0}[x^2]$ and $\mu_1 := \mathbb{E}_{y \leftarrow \mathcal{D}_1}[y^2]$. Note that we have

$$e = |\mu_0 - \mu_1|. \tag{2.229}$$

According to Theorem 70, we have

$$\Pr\left[\left|\overline{\beta} - \mu_1\right| \geq \frac{e}{4}\right] = \Pr\left[\left|\frac{\overline{\beta}}{B^2} - \frac{\mu_1}{B^2}\right| \geq \frac{e}{4B^2}\right] \leq 2 \cdot \exp\left(-\frac{N \cdot e^2}{32 \cdot B^4}\right), \tag{2.230}$$

$$\Pr\left[\left|\overline{\gamma} - \mu_0\right| \geq \frac{e}{4}\right] = \Pr\left[\left|\frac{\overline{\gamma}}{B^2} - \frac{\mu_0}{B^2}\right| \geq \frac{e}{4B^2}\right] \leq 2 \cdot \exp\left(-\frac{N \cdot e^2}{32 \cdot B^4}\right). \tag{2.231}$$

By a union bound, the inequalities $\left|\overline{\beta} - \mu_1\right| < \frac{e}{4}$ and $|\overline{\gamma} - \mu_0| < \frac{e}{4}$ hold simultaneously with probability $> 1 - 4\exp\left(-N \cdot e^2/\left(32 \cdot B^4\right)\right)$. By using triangle inequalities, the following inequality will hold with the same probability

$$\left|\overline{\beta} - \overline{\gamma}\right| \geq |\mu_0 - \mu_1| - |\overline{\gamma} - \mu_0| - \left|\overline{\beta} - \mu_1\right| > e - e/4 - e/4 = e/2. \qquad \square$$

Lemma 73. *Denote the distinguisher from Algorithm 9 by $\mathcal{T}$. Let $\mathcal{D}_0$, $\mathcal{D}_1$ be two B-bounded distributions. Let $N' \in \mathbb{N}$ and set*

$$r = 4B^2/N' \qquad \text{and} \qquad N = {N'}^3. \tag{2.232}$$

1. *We have*

$$\mathsf{adv}^{\mathsf{Dist}}_{\mathcal{D}_0,\mathcal{D}_1,N,r}(\mathcal{T}) \geq -8\exp(-2N'). \tag{2.233}$$

2. *If $N' \geq 8B^2/e$ for*

$$e = \left|\mathop{\mathbb{E}}_{x \leftarrow \mathcal{D}_0}[x^2] - \mathop{\mathbb{E}}_{y \leftarrow \mathcal{D}_1}[y^2]\right|, \tag{2.234}$$

we have

$$\mathsf{adv}^{\mathsf{Dist}}_{\mathcal{D}_0,\mathcal{D}_1,N,r}(\mathcal{T}) \geq 1 - 16\exp(-2N'). \tag{2.235}$$

Proof. 1. Let $\alpha_1, \ldots, \alpha_N, \gamma_1, \ldots, \gamma_N \leftarrow \mathcal{D}_0$ and $\beta_1, \ldots, \beta_N \leftarrow \mathcal{D}_1$ be the input for $\mathcal{T}$. Let $e_0 = |\overline{\alpha} - \overline{\gamma}|$ be the value computed by $\mathcal{T}$ in Line 4. According to Lemma 71, we have

$$\Pr[e_0 > r] < 4\exp(-2N'). \tag{2.236}$$

Hence, the probability that the if-check in Line 6 on input $(\alpha_i)_i, (\beta_i)_i, (\gamma_i)_i$, r passes is at most $4\exp(-2N')$. If the check in Line 6 does not pass, $\mathcal{T}$ will output 0 with probability at least 1/2. It follows

$$\Pr[\mathcal{T}((\alpha_i)_i, (\beta_i)_i, (\gamma_i)_i, r) = 0] \geq \frac{1}{2} - 4\exp(-2N'). \tag{2.237}$$

Because of symmetry, we get for $\gamma_1, \ldots, \gamma_N \leftarrow \mathcal{D}_1$

$$\Pr[\mathcal{T}((\alpha_i)_i, (\beta_i)_i, (\gamma_i)_i, r) = 1] \geq \frac{1}{2} - 4\exp(-2N'). \tag{2.238}$$

It follows

$$\mathsf{adv}^{\mathsf{Dist}}_{\mathcal{D}_0,\mathcal{D}_1,N,r}(\mathcal{T}) \geq -8\exp(-2N'). \tag{2.239}$$

2. Again, let $\alpha_1, \dots, \alpha_N, \gamma_1, \dots, \gamma_N \leftarrow \mathcal{D}_0$ and $\beta_1, \dots, \beta_N \leftarrow \mathcal{D}_1$. We claim that the if-check in Line 8 will be passed with probability at least $1 - 8\exp(-2N')$.

   For $e_0 = |\overline{\alpha} - \overline{\gamma}|$, we have

$$\Pr[e_0 < r] > 1 - 4\exp(-2N'), \tag{2.240}$$

   because of Lemma 71. Because of Lemma 72, we have

$$\Pr[e_1 > e/2] > 1 - 4\exp(-Ne^2/(32B^4)) \tag{2.241}$$

   where $e_1 = \left|\overline{\beta} - \overline{\gamma}\right|$ is the intermediate value computed by $\mathcal{T}$. Since $r = 4B^2/N'$ and $N' \geq 8B^2/e$, we have $r \leq e/2$. Hence, Eq. (2.241) implies

$$\begin{aligned} \Pr[e_1 > r] >& 1 - 4\exp(-Ne^2/(32B^4)) && (2.242)\\ =& 1 - 4\exp(-{N'}^3 e^2/32B^4) && (2.243)\\ \geq& 1 - 4\exp(-N' \cdot 8^2 B^4 e^2/(32B^4e^2)) && (2.244)\\ =& 1 - 4\exp(-2N') && (2.245)\end{aligned}$$

   By a union bound, we now get

$$\Pr[e_1 > r > e_0] > 1 - 8\exp(-2N'). \tag{2.246}$$

   Hence, we have

$$\Pr\left[\mathcal{T}((\alpha_i)_i, (\beta_i)_i, (\gamma_i)_i, r) = 0\right] \geq 1 - 8\exp(-2N'). \tag{2.247}$$

   Because of symmetry, we get for $\gamma_1, \dots, \gamma_N \leftarrow \mathcal{D}_1$

$$\Pr\left[\mathcal{T}((\alpha_i)_i, (\beta_i)_i, (\gamma_i)_i, r) = 1\right] \geq 1 - 8\exp(-2N'). \tag{2.248}$$

   It follows

$$\mathsf{adv}^{\mathsf{Dist}}_{\mathcal{D}_0, \mathcal{D}_1, N, r}(\mathcal{T}) \geq 1 - 16\exp(-2N'). \qquad \square$$

In general, Algorithm 9 does not need to be a strong distinguisher. Lemma 73 guarantees that its advantage will be close to zero for r large enough. However, we can only assume a large distinguishing advantage for Algorithm 9 if the squared means of $\mathcal{D}_0$ and $\mathcal{D}_1$ differ by a non-negligible amount. The next lemma shows that this is indeed the case if $\mathcal{D}_0$ and $\mathcal{D}_1$ both stem from a distribution of univariate polynomials of constant degree.

Lemma 74. *Let $d, N' \in \mathbb{N}$ and $B \geq 1/2$. Further, let $\mathcal{E}$ be a memoryless distribution of integer univariate polynomials of degree d, i.e., the support of $\mathcal{E}$ lies in $\mathbb{Z}[X]^{\leq d}$. Set*

$$p := \Pr_{f \leftarrow \mathcal{E}}[\deg f > 0]. \tag{2.249}$$

For $x \in \{0, \dots, 2d\}$, denote by $\mathcal{D}_x$ the distribution that samples $f \leftarrow \mathcal{E}$ and outputs $f(x)$. Assume that we have:

1. $N' \geq 8 \cdot \Gamma_{2d} \cdot B^2(1 + 2B \cdot \Gamma_d)^d/p$,

2. $r = 4B^2/N'$,

3. $N = N'^3$.

If $\mathcal{D}_0, \mathcal{D}_1, \ldots, \mathcal{D}_{2d}$ *are bounded by* B*, then there is an* $x_\dagger \in [2d]$ *s.t.*

$$\mathsf{adv}^{\mathsf{Dist}}_{\mathcal{D}_0,\mathcal{D}_{x_\dagger},N,r}(\mathcal{T}) \geq 1 - 16\exp(-2N'). \tag{2.250}$$

For each other $x \in [2d]$*, we have*

$$\mathsf{adv}^{\mathsf{Dist}}_{\mathcal{D}_0,\mathcal{D}_x,N,r}(\mathcal{T}) \geq -8\exp(-2N'). \tag{2.251}$$

Proof. Eq. (2.251) follows from the first part of Lemma 73.

To prove Eq. (2.250), set e to be the maximum of $\left|\mathbb{E}[f(x)^2 - f(0)^2]\right|$, i.e.,

$$e := \max_{x \in [2d]} \left|\mathbb{E}[f(x)^2 - f(0)^2]\right|, \tag{2.252}$$

and let $x_\dagger \in [2d]$ be s.t. $e = \left|\mathbb{E}[f(x_\dagger)^2 - f(0)^2]\right|$. According to Theorem 55, we have

$$p \leq \Gamma_{2d} \cdot e \cdot (1 + 2B \cdot \Gamma_d)^d. \tag{2.253}$$

This lower-bounds e by

$$N' \geq 8 \cdot \Gamma_{2d} \cdot B^2(1 + 2B \cdot \Gamma_d)^d/p \geq 8B^2/e. \tag{2.254}$$

By Lemma 73, it follows now

$$\mathsf{adv}^{\mathsf{Dist}}_{\mathcal{D}_0,\mathcal{D}_{x_\dagger},N,r}(\mathcal{T}) \geq 1 - 16\exp\left(-N'\right). \tag{2.255}$$

Hence, Eq. (2.250) is proven. □

In conclusion, we could show that Algorithm 9 can successfully distinguish between $f(0)$ and $f(x)$ for random univariate polynomials $f \leftarrow \mathcal{E}$ for some $x \in [2d]$ when it takes

$$N \in \Omega(B^{6+3d}/p^3) \tag{2.256}$$

samples where B is a bound for $|f(x)|$, $d = \deg f$ and p is the probability that f is non-constant. Note that p is an upper bound for the statistical distance between the distributions $f(0)$ and $f(x)$. Hence, if x can be extracted from $f(x)$, it must follow that Algorithm 9 has an overwhelming advantage at distinguishing $f(0)$ from $f(x)$ while requiring a number of samples that is polynomial in B. This will lead to the selective IND-CPA adversary for SKEs based on random polynomials that we will introduce in the next section.

Remark 6. It would be interesting to simplify Algorithm 9. Concretely, one could remove the control parameter r from Algorithm 9 and instead consider a simplified version, for which one can still prove an overwhelming distinguishing advantage if

$$\left|\mathop{\mathbb{E}}_{x \leftarrow \mathcal{D}_0}[x^2] - \mathop{\mathbb{E}}_{y \leftarrow \mathcal{D}_1}[y^2]\right| \tag{2.257}$$

is large enough (i.e., non-negligible). Removing the check for r would concretely yield the following algorithm:

```
set $\overline{\alpha} := \frac{1}{N}\sum_{i=1}^{N}\alpha_i^2$
set $\overline{\beta} := \frac{1}{N}\sum_{i=1}^{N}\beta_i^2$
set $\overline{\gamma} := \frac{1}{N}\sum_{i=1}^{N}\gamma_i^2$
set $e_0 := |\overline{\alpha} - \overline{\gamma}|$
set $e_1 := |\overline{\beta} - \overline{\gamma}|$
if $e_0 > e_1$ then
    return $1$
else if $e_1 > e_0$ then
    return $0$
else
    draw $b' \leftarrow \{0,1\}$
    return $b'$
end if
```

Unfortunately, it turns out to be very complicated to prove a good lower bound for the advantage of the above algorithm if $\mathcal{D}_0$ and $\mathcal{D}_1$ are arbitrary bounded distributions. To prove that the distinguishing advantage of this algorithm is always non-negative, it would suffice to prove the following inequality for arbitrary discrete distributions $\overline{\mathcal{D}}_0$ and $\overline{\mathcal{D}}_1$ over $[0, B]$

$$\Pr_{\substack{\overline{\alpha},\overline{\gamma}\leftarrow\overline{\mathcal{D}}_0\\ \overline{\beta}\leftarrow\overline{\mathcal{D}}_1}}\left[|\overline{\alpha}-\overline{\gamma}| < |\overline{\beta}-\overline{\gamma}|\right] + \frac{1}{2}\Pr_{\substack{\overline{\alpha},\overline{\gamma}\leftarrow\overline{\mathcal{D}}_0\\ \overline{\beta}\leftarrow\overline{\mathcal{D}}_1}}\left[|\overline{\alpha}-\overline{\gamma}| = |\overline{\beta}-\overline{\gamma}|\right] \tag{2.258}$$

$$\geq \Pr_{\substack{\overline{\alpha}\leftarrow\overline{\mathcal{D}}_0\\ \overline{\beta},\overline{\gamma}\leftarrow\overline{\mathcal{D}}_1}}\left[|\overline{\alpha}-\overline{\gamma}| < |\overline{\beta}-\overline{\gamma}|\right] + \frac{1}{2}\Pr_{\substack{\overline{\alpha}\leftarrow\overline{\mathcal{D}}_0\\ \overline{\beta},\overline{\gamma}\leftarrow\overline{\mathcal{D}}_1}}\left[|\overline{\alpha}-\overline{\gamma}| = |\overline{\beta}-\overline{\gamma}|\right]. \tag{2.259}$$

While experiments suggest the correctness of this inequality for discrete distributions over the set $\{0, \ldots, 20\}$, it seems to be complicated to prove this inequality, given its interplay of stochastic and geometric structure.

For discrete distributions $\overline{\mathcal{D}}_0$, $\overline{\mathcal{D}}_1$ over finite sets $\{0, \ldots, n\}$, the inequality can be shown to be equivalent to the positive semi-definiteness of certain quadrilinear resp. bilinear forms. Unfortunately, these forms do not admit an easily understandable structure, and I was unable to prove their semi-definiteness. Hence, the correctness of Eq. (2.259) remains an open question.

Question 7. *Does Eq.* (2.259) *do hold for each $n \in \mathbb{N}$ and all discrete distributions $\overline{\mathcal{D}}_0$, $\overline{\mathcal{D}}_1$ with support in the $\{0, \ldots, n\}$?*

2.4 On Secret-Key Encryption

In this section, we want to prove the following theorem:

Theorem 75. *Let $\mathsf{SKE} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{Dec})$ be an SKE for messages $\mathcal{X} \subset \mathbb{Z}$ and let $q = q(\lambda)$ be prime. Assume that the following requirements are met:*

1. *The message space $\mathcal{X}$ contains the numbers $0, \ldots, 2d$.*
2. *SKE is of depth d over $\mathbb{Z}_q$ and each ciphertext lies in $\mathbb{Z}_q^m$ for $m \in \mathsf{poly}(\lambda)$.*
3. *SKE is of width B over $\mathbb{Z}_q$ with a function $\varepsilon_{\mathsf{width}} \in \mathsf{negl}(\lambda)$ s.t. we have for all $(x_\lambda)_\lambda \in \mathcal{X}$*

$$\Pr_{\substack{\mathsf{msk}\leftarrow\mathsf{Setup}(1^\lambda)\\ \mathsf{ct}\leftarrow\mathsf{Enc}(\mathsf{msk},x_\lambda)}}\left[||\mathsf{ct}||_\infty > B\right] \leq \varepsilon_{\mathsf{width}}(\lambda). \tag{2.260}$$

4. *The inequality*

$$2 \cdot (d+1)! \cdot (2d)^d \cdot \Gamma_d \cdot B < q \tag{2.261}$$

holds where Γ_d denotes the interpolation number from Theorem 54.

Let $N' \in \mathbb{N}$ s.t.

$$N' \geq \frac{16 \cdot (d!)^2 \cdot \Gamma_{2d} \cdot m \cdot B^2 \cdot (1 + 2 \cdot d! \cdot \Gamma_d \cdot B)^d}{\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}) - (d+2)\varepsilon_{\mathsf{width}}}. \tag{2.262}$$

There is an adversary $\mathcal{A}$ against the selective IND-CPA security of SKE *that makes $3{N'}^3$ encryption queries, $\Theta(m{N'}^3)$ arithmetical operations over $\mathbb{Z}$ and $\mathbb{Z}_q$ and has an advantage of*

$$\geq \frac{\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}) - (d+2)\varepsilon_{\mathsf{width}}}{4dm} - 16\exp(-2N') - (d+4) \cdot N \cdot \varepsilon_{\mathsf{width}}.$$

Remark 7. Note that in Theorem 75 we only bound the number of arithmetic operations over $\mathbb{Z}$ and $\mathbb{Z}_q$ made by $\mathcal{A}$. In general, arithmetic operations over $\mathbb{Z}$ may require superpolynomial many bit operations. Hence, it is from the claim of Theorem 75 not clear if $\mathcal{A}$ can be translated into a PPT adversary for $B \in \mathsf{poly}(\lambda)$.

As we will see later, the operations over $\mathbb{Z}$ stem from the mean square distinguisher $\mathcal{T}$ from Algorithm 9. Since the input for $\mathcal{T}$ will always be bounded by q, Lemma 69 states that the bit complexity of $\mathcal{T}$ is bounded by

$$O(\log(q)^2 \cdot \log(N) \cdot N). \tag{2.263}$$

(This is, because $\mathcal{T}$'s time complexity is dominated by computing the sum of N squares of its inputs.)

Hence, the adversary $\mathcal{A}$ can be implemented such that it makes

$$O(N \cdot \log(q) \cdot (m + \log(q) \cdot \log(N))) \tag{2.264}$$

bit operations (where we ignore the problem of uniformly sampling random numbers).

We can immediately deduce the following corollary from Theorem 77:

Corollary 76. *In the situation of Theorem 75, if*

$$\varepsilon_{\mathsf{width}} \in \mathsf{negl}(\lambda) \qquad \textit{and} \qquad \mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}) \notin \mathsf{negl}(\lambda), \tag{2.265}$$

then there is an attack against the selective IND-CPA security of SKE *that has a non-negligible advantage, makes* $\mathsf{poly}(\lambda + B)$ *arithmetic operations over $\mathbb{Z}_q$ and $\mathbb{Z}$ and queries* $\mathsf{poly}(\lambda + B)$ *many ciphertexts.*

Proof. Since $\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}) \notin \mathsf{negl}(\lambda)$, there is an $s \in \mathsf{poly}(\lambda)$ and an infinitely large $\Lambda \subset \mathbb{N}$ s.t.

$$\forall \lambda \in \Lambda : \quad \mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}) > 1/s(\lambda). \tag{2.266}$$

We run the attack from Theorem 75 with

$$N' := \max\left(\lambda, \left\lceil \frac{16 \cdot (d!)^2 \cdot \Gamma_{2d} \cdot m \cdot B^2 \cdot (1 + 2 \cdot d! \cdot \Gamma_d \cdot B)^d}{s(\lambda)^{-1} - (d+2)\varepsilon_{\mathsf{width}}} \right\rceil\right). \tag{2.267}$$

Whenever $\lambda \in \Lambda$, the advantage of this adversary is at least

$$\frac{1}{4d \cdot m(\lambda) \cdot s(\lambda)} - \mathsf{negl}(\lambda), \tag{2.268}$$

which is non-negligible, since Λ is of finite size. □

To prove Theorem 75, we will describe in Section 2.4.1 an attack on SKEs of polynomial width and constant depth over $\mathbb{Z}$. In Section 2.4.2, we will—by a cryptographic reduction—extend this attack to SKEs over $\mathbb{Z}_q$ of small width and constant depth. This will prove Theorem 75.

Convention 3. To simplify notation in this section, we will sometimes write r or $r(X)$ instead of $(r_1, \ldots, r_m)$ when we draw a tuple of polynomials $(r_1, \ldots, r_m) \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$ from an offline algorithm. Further, for a message $x \in \mathbb{Z}^n$, we will write $r(x)$ instead of $(r_1(x), \ldots, r_m(x)) \in \mathbb{Z}^m$, and we will set

$$\deg r := \max_{i \in [m]} \deg r_i. \tag{2.269}$$

2.4.1 SKE over $\mathbb{Z}$

For this subsection, let $\mathsf{SKE} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{Dec})$ be an SKE for integer messages $\mathcal{X} \subset \mathbb{Z}$ s.t. for some constant $d \in \mathbb{N}$

$$\{0, \ldots, 2d\} \subseteq \mathcal{X}. \tag{2.270}$$

Assume that ciphertexts of SKE lie in $\mathbb{Z}^m$ for some parameter $m \in \mathsf{poly}(\lambda)$. Additionally, assume for this subsection that the encryption algorithm Enc of SKE is of depth d. I.e., there is an algorithm $\mathsf{Enc}_{\mathsf{off}}$ that on input a master secret key msk outputs m polynomials $r_1, \ldots, r_m \in \mathbb{Z}[X]^{\leq d}$. The polynomials $r_1, \ldots, r_m$ are univariate and of constant degree d. They are sampled randomly by $\mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$, but may be correlated with each other. Now, Enc encrypts messages $x \in \mathcal{X}$ under msk, by sampling $(r_1, \ldots, r_m) \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$ and outputting

$$\mathsf{ct} := (r_1(x), \ldots, r_m(x)). \tag{2.271}$$

Additionally, assume that Enc is of width $B > 0$ over $\mathbb{Z}$. I.e., there is a negligible function $\varepsilon_{\mathsf{width}} \in \mathsf{negl}(\lambda)$ s.t. we have for each $\lambda \in \mathbb{N}$ and each $x_\lambda \in \mathcal{X}_\lambda$

$$\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda)}} [||\mathsf{ct}||_\infty > B] \leq \varepsilon_{\mathsf{width}}. \tag{2.272}$$

The main goal in this subsection is to prove that such a scheme SKE cannot be selectively IND-CPA secure. For this end, we give the following adversary against SKE, for which we will prove a non-negligible advantage in Theorem 77.

Algorithm 10. We will describe here an adversary $\mathcal{A}$ that plays the selective IND-CPA security Game 3 of SKE. $\mathcal{A}$ is controlled by three parameters $d, B, N' \in \mathbb{N}$. It proceeds in the following steps:

Step 1: $\mathcal{A}$ computes the parameters

$$N := N'^3 \qquad \text{and} \qquad r := 4B^2/N'. \tag{2.273}$$

Step 2: $\mathcal{A}$ draws a random message $x_* \leftarrow [2d]$ and, for $i \in [N]$, sets

$$x_i^{(0)} := 0, \qquad x_i^{(1)} := 0, \tag{2.274}$$

$$x_{N+i}^{(0)} := x_*, \qquad x_{N+i}^{(1)} := x_*, \tag{2.275}$$

$$x_{2N+i}^{(0)} := 0, \qquad x_{2N+i}^{(1)} := x_*. \tag{2.276}$$

Step 3: $\mathcal{A}$ submits the two list of messages $(x_i^{(0)})_{i=1}^{3N}, (x_i^{(1)})_{i=1}^{3N}$ to the challenger $\mathcal{C}$ and receives a list of ciphertexts $\mathsf{ct}_1, \ldots, \mathsf{ct}_{3N} \in \mathbb{Z}^m$.

Step 4: $\mathcal{A}$ draws a uniformly random index $j_* \leftarrow [m]$. Denote by $\pi_{j_*} : \mathbb{Z}^m \to \mathbb{Z}$ the projection to the j_*-th coordinate. For $i \in [N]$, $\mathcal{A}$ sets

$$\alpha_i := \pi_{j_*}(\mathsf{ct}_i), \tag{2.277}$$

$$\beta_i := \pi_{j_*}(\mathsf{ct}_{N+i}), \tag{2.278}$$

$$\gamma_i := \pi_{j_*}(\mathsf{ct}_{2N+i}). \tag{2.279}$$

Step 5: Denote by $\mathcal{T}$ the mean square distinguisher from Algorithm 9. $\mathcal{A}$ runs

$$b' \leftarrow \mathcal{T}((\alpha_i)_i, (\beta_i)_i, (\gamma_i)_i, r) \tag{2.280}$$

and outputs b'.

Theorem 77. *Let*

1. $p := \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})}} [\deg r > 0]$,
2. $N' \geq \left\lceil \frac{16 \cdot \Gamma_{2d} \cdot m \cdot B^2 \cdot (1 + 2B \cdot \Gamma_d)^d}{p - \varepsilon_{\mathsf{width}}} \right\rceil$,
3. $N := N'^3$.

The adversary from Algorithm 10 instantiated with d, B and N' has an advantage of at least

$$\mathsf{adv}_{\mathsf{SKE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}) \geq \frac{p - \varepsilon_{\mathsf{width}}}{4dm} - 16 \exp(-2N') - 3N \cdot \varepsilon_{\mathsf{width}} \tag{2.281}$$

in the selective IND-CPA security Game 3 of SKE*. It makes*

$$\Theta(mN) = \Theta\left(\frac{m^4 \cdot B^{6+3d}}{(p - \varepsilon_{\mathsf{width}})^3} \right) \tag{2.282}$$

arithmetical operations over $\mathbb{Z}$ and queries $3N$ ciphertexts from the challenger of Game 3.

Corollary 78. *If*

$$\varepsilon \in \mathsf{negl}(\lambda) \qquad \textit{and} \qquad \mathsf{adv}_{\mathsf{SKE}}^{\mathsf{dec}}(\mathsf{Dec}) \notin \mathsf{negl}(\lambda), \tag{2.283}$$

then Algorithm 10 gives an adversary with non-negligible advantage against the selective IND-CPA security of SKE *that makes $\mathsf{poly}(\lambda + B)$ encryption queries and arithmetic operations over $\mathbb{Z}$.*

Proof. Because of Lemma 52, we have

$$p(\lambda) = \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})}} [\deg r > 0] \geq \mathsf{adv}_{\mathsf{SKE}}^{\mathsf{dec}}(\mathsf{Dec}) \notin \mathsf{negl}(\lambda). \tag{2.284}$$

Hence, there is an $s \in \mathsf{poly}(\lambda)$ and an infinite subset $\Lambda \subset \mathbb{N}$ s.t.

$$\forall \lambda \in \Lambda : \quad p(\lambda) > \frac{1}{s(\lambda)}. \tag{2.285}$$

Set

$$N' := \left\lceil \frac{16 \cdot \Gamma_{2d} \cdot m \cdot B^2 \cdot (1 + 2B \cdot \Gamma_d)^d}{s^{-1} - \varepsilon_{\mathsf{width}}} \right\rceil \tag{2.286}$$

Theorem 77 implies that Algorithm 10 instantiated with d, B and N' is an algorithm of time $\mathsf{poly}(B + \lambda)$ whose advantage is at least $\frac{1}{4dms} - \mathsf{negl}(\lambda)$ whenever $\lambda \in \Lambda$. □

In the following, we will prove Theorem 77. For this end, we will need to show Lemmas 79 to 81:

Lemma 79. *Let $\mathcal{A}$ be an adversary for Game 3 of* SKE *and let* event *denote some event that occurs during $\mathcal{A}$'s run in Game 3 with non-zero probability. Let $b \leftarrow \{0,1\}$ be the random bit drawn by the challenger in Game 3 and denote by b' the response of $\mathcal{A}$. Further, denote by* $\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A} \mid \mathsf{event})$ *the advantage of $\mathcal{A}$ in all runs of Game 3 where* event *did occur, i.e.*

$$\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A} \mid \mathsf{event}) := 2 \cdot \Pr[b = b' | \mathsf{event}] - 1. \tag{2.287}$$

Then, $\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A})$ *equals*

$$\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A} \mid \mathsf{event}) \cdot \Pr[\mathsf{event}] + \mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A} \mid \neg\mathsf{event}) \cdot \Pr[\neg\mathsf{event}].$$

Proof. We have

$$\begin{aligned}
&\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A}) = 2 \cdot \Pr[b = b'] - 1 \\
=&2 \cdot (\Pr[b = b' \mid \mathsf{event}] \cdot \Pr[\mathsf{event}] + \Pr[b = b' \mid \neg\mathsf{event}] \cdot \Pr[\neg\mathsf{event}]) - 1 \\
=&(2 \cdot \Pr[b = b' \mid \mathsf{event}] - 1) \cdot \Pr[\mathsf{event}] + (2 \cdot \Pr[b = b' \mid \neg\mathsf{event}] - 1) \cdot \Pr[\neg\mathsf{event}] \\
=&\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A} \mid \mathsf{event}) \cdot \Pr[\mathsf{event}] + \mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A} \mid \neg\mathsf{event}) \cdot \Pr[\neg\mathsf{event}]. \qquad \square
\end{aligned}$$

Lemma 80. *Let $\mathcal{A}$ be the adversary of Algorithm 10 instantiated with d, B and*

$$N' \geq \left\lceil \frac{16 \cdot \Gamma_{2d} \cdot m \cdot B^2 \cdot (1 + 2B \cdot \Gamma_d)^d}{p} \right\rceil \tag{2.288}$$

for

$$p = \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})}} [\deg r > 0] > 0. \tag{2.289}$$

If $\varepsilon_{\mathsf{width}} = 0$, then the advantage of $\mathcal{A}$ in the selective IND-CPA security Game 3 of SKE *is lower-bounded by*

$$\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A}) \geq \frac{p}{4dm} - 16\exp(-2N') \tag{2.290}$$

Proof. We want to lower-bound the advantage of $\mathcal{A}$, which equals the advantage of the mean square distinguisher $\mathcal{T}$ from Algorithm 9 it uses in Step 5. We will call a master secret key msk *good* iff

$$\Pr_{(r_1,\ldots,r_m) \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})} [\exists j \in [m] : \deg r_j > 0] \geq \frac{p}{2}. \tag{2.291}$$

According to Lemma 51, we have

$$\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)}[\mathsf{msk} \text{ is good}] > \frac{p}{2}. \tag{2.292}$$

Let $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$ be the master secret key drawn by $\mathcal{C}$. Assume that msk is good. In this case, if we draw $r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$, we have $\deg r > 0$ with probability at least $p/2$. Now, for each $\lambda \in \mathbb{N}$, there will be one $j_\dagger \in [m]$ (that depends on msk) s.t.

$$p_{j_\dagger} := \Pr_{r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})}[\deg r_{j_\dagger} > 0] \geq \frac{p}{2m}. \tag{2.293}$$

The probability that the index $j_* \leftarrow [m]$ drawn by $\mathcal{A}$ in Step 4 equals $j_\dagger$ is $1/m$. Since $p_{j_\dagger} \geq \frac{p}{2m}$, it follows $N' \geq 8\Gamma_{2d} \cdot B^2(1 + 2B \cdot \Gamma_d)^d / p_{j_\dagger}$. Hence, Lemma 74 implies that there is one $x_\dagger \in [2d]$ s.t.

$$\mathsf{adv}^{\mathsf{Dist}}_{\mathcal{D}_0,\mathcal{D}_1,N,r}(\mathcal{T}) \geq 1 - 16\exp(-2N') \tag{2.294}$$

where $\mathcal{D}_0$ and $\mathcal{D}_1$ sample $r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$ and output $r_{j_\dagger}(0)$ and $r_{j_\dagger}(x_\dagger)$, respectively. Since $\varepsilon_{\mathsf{width}} = 0$, $\mathcal{D}_0$ and $\mathcal{D}_1$ are bounded by B. The message $x_* \leftarrow [2d]$ drawn by $\mathcal{A}$ in Step 2 equals $j_\dagger$ with probability $1/(2d)$. If $x_* = x_\dagger$ and $j_* = j_\dagger$, then the values $\alpha_i = \pi_{j_\dagger}(\mathsf{ct}_i)$, $i \in [N]$, will be distributed according to $\mathcal{D}_0$, the values $\beta_i = \pi_{j_\dagger}(\mathsf{ct}_{i+N})$, $i \in [N]$, will be distributed according to $\mathcal{D}_1$, and the values $\gamma_i = \pi_{j_\dagger}(\mathsf{ct}_{i+2N})$ will be distributed according to $\mathcal{D}_b$, where b is the secret bit chosen by the challenger in Game 3. In this case ($x_* = x_\dagger$, $j_* = j_\dagger$ and msk is good), $\mathcal{A}$'s advantage will equal the advantage of $\mathcal{T}$ in Eq. (2.294). Hence, we have

$$\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A} \mid \mathsf{msk} \text{ is good}, x_* = x_\dagger, j_* = j_\dagger) \geq 1 - 16\exp(-2N'). \tag{2.295}$$

In each other case, the advantage of $\mathcal{T}$ (and hence the advantage of $\mathcal{A}$) will be lower-bounded by $-8\exp(-2N')$, according to Lemma 73. Hence,

$$\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A} \mid \mathsf{msk} \text{ is not good}, x_* \neq x_\dagger \text{ or } j_* \neq j_\dagger) \geq -8\exp(-2N'). \tag{2.296}$$

By Lemma 79, we have

$$\begin{aligned}
&\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A}) && (2.297)\\
=&\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A} \mid \mathsf{msk} \text{ is good}, x_* = x_\dagger, j_* = j_\dagger) && (2.298)\\
&\cdot \Pr[\mathsf{msk} \text{ is good}, x_* = x_\dagger, j_* = j_\dagger] && (2.299)\\
&+ \mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A} \mid \mathsf{msk} \text{ is not good}, x_* \neq x_\dagger \text{ or } j_* \neq j_\dagger) && (2.300)\\
&\cdot \Pr[\mathsf{msk} \text{ is not good}, x_* \neq x_\dagger \text{ or } j_* \neq j_\dagger] && (2.301)\\
\geq&(1 - 16\exp(-2N')) \cdot \Pr[\mathsf{msk} \text{ is good}, x_* = x_\dagger, j_* = j_\dagger] && (2.302)\\
&+ (-8\exp(-2N')) \cdot \Pr[\mathsf{msk} \text{ is not good}, x_* \neq x_\dagger \text{ or } j_* \neq j_\dagger] && (2.303)\\
>&(1 - 16\exp(-2N')) \cdot \frac{p}{2} \cdot \frac{1}{2d} \cdot \frac{1}{m} - 8\exp(-2N')\left(1 - \frac{p}{2} \cdot \frac{1}{2d} \cdot \frac{1}{m}\right) && (2.304)\\
>&(1 - 16\exp(-2N')) \cdot \frac{p}{2} \cdot \frac{1}{2d} \cdot \frac{1}{m} - 16\exp(-2N')\left(1 - \frac{p}{2} \cdot \frac{1}{2d} \cdot \frac{1}{m}\right) && (2.305)\\
\geq&\frac{p}{4dm} - 16\exp(-2N'). && \square
\end{aligned}$$

Algorithm 11. We define here an alternative offline algorithm $\mathsf{Enc}'_{\mathsf{off}}$ for the encryption of SKE:

$\mathsf{Enc}'_{\mathsf{off}}$: On input a master secret key msk, $\mathsf{Enc}_{\mathsf{off}}$ samples

$$(r_1, \ldots, r_m) \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}). \tag{2.306}$$

If there is a $j \in [m]$ and an $x \in \mathcal{X}_\lambda$ s.t. $|r_j(x)| > B$, then $\mathsf{Enc}'_{\mathsf{off}}$ outputs m zero polynomials $(0, \ldots, 0)$. Otherwise, it outputs $(r_1, \ldots, r_m)$.

Lemma 81. *The statistical distance between the distribution*

$$(\mathsf{msk}, r^{(1)}, \ldots, r^{(3N)}), \tag{2.307}$$

for $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$*,* $r^{(1)}, \ldots, r^{(3N)} \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$*, and the distribution*

$$(\mathsf{msk}', r'^{(1)}, \ldots, r'^{(3N)}), \tag{2.308}$$

for $\mathsf{msk}' \leftarrow \mathsf{Setup}(1^\lambda)$*,* $r'^{(1)}, \ldots, r'^{(3N)} \leftarrow \mathsf{Enc}'_{\mathsf{off}}(\mathsf{msk}')$*, is bounded by*

$$3N \cdot \varepsilon_{\mathsf{width}}(\lambda). \tag{2.309}$$

Proof. Because of Lemma 50, the statistical distance between (msk, r), for $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$ and $r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$, and (msk', r'), for $\mathsf{msk}' \leftarrow \mathsf{Setup}(1^\lambda)$ and $r' \leftarrow \mathsf{Enc}'_{\mathsf{off}}(\mathsf{msk}')$, is bounded by $\varepsilon_{\mathsf{width}}$. Lemma 50 now implies

$$\Delta\Big((\mathsf{msk}, r^{(1)}, \ldots, r^{(3N)}), (\mathsf{msk}', r'^{(1)}, \ldots, r'^{(3N)})\Big) \leq 3N \cdot \varepsilon_{\mathsf{width}}(\lambda). \qquad \square$$

Proof Theorem 77. Denote by $\mathsf{Enc}'_{\mathsf{off}}$ the alternative offline algorithm from Algorithm 11 and let Enc' be an alternative encryption algorithm for SKE that samples $r \leftarrow \mathsf{Enc}'_{\mathsf{off}}(\mathsf{msk})$ and outputs $\mathsf{ct} = r(x)$ as ciphertext on input a message $x \in \mathcal{X}$ and a master secret key msk. We can now consider the alternative scheme $\mathsf{SKE}' = (\mathsf{Setup}, \mathsf{Enc}', \mathsf{Dec})$.

For each master secret key msk and each message $x \in \mathcal{X}_\lambda$, the ciphertext $\mathsf{ct} \leftarrow \mathsf{Enc}'(\mathsf{msk}, x)$ will always be bounded by B. Further, we have

$$\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ r \leftarrow \mathsf{Enc}'_{\mathsf{off}}(\mathsf{msk})}} [\deg r > 0] \geq \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})}} [\deg r > 0] - \varepsilon_{\mathsf{width}} \geq p - \varepsilon_{\mathsf{width}}. \tag{2.310}$$

Lemma 80 now implies that the advantage of the adversary from Algorithm 10 instantiated with d, B and

$$N' \geq \left\lceil \frac{16 \cdot \Gamma_{2d} \cdot m \cdot B^2 \cdot (1 + 2B \cdot \Gamma_d)^d}{p - \varepsilon_{\mathsf{width}}} \right\rceil \tag{2.311}$$

against SKE' is at least

$$\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}'}(\mathcal{A}) \geq \frac{p - \varepsilon_{\mathsf{width}}}{4dm} - 16 \exp(-2N'). \tag{2.312}$$

Lemma 81 shows that the statistical distance of the view of Algorithm 10 when playing against SKE' and SKE is bounded by $3N \cdot \varepsilon_{\mathsf{width}}$. It follows,

$$\begin{aligned} \mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A}) &\geq \mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}'}(\mathcal{A}) - 3N \cdot \varepsilon_{\mathsf{width}} && (2.313) \\ &\geq \frac{p - \varepsilon_{\mathsf{width}}}{4dm} - 16 \exp(-2N') - 3N \cdot \varepsilon_{\mathsf{width}}. && (2.314) \end{aligned}$$

This finishes the proof of Theorem 77. $\square$

2.4.2 From $\mathbb{Z}$ to $\mathbb{Z}_q$

In this subsection, let $n = n(\lambda)$ be a parameter and define for $\lambda \in \mathbb{N}$

$$P_\lambda := \left\{ x \in \mathbb{N}_0^{n(\lambda)} \;\middle|\; d \geq ||x||_1 \right\} \tag{2.315}$$

Note that P_λ is the standard point basis for degree-d polynomials over n variables that we discussed in Section 2.2. Let $\mathsf{SKE} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{Dec})$ be an SKE for messages $\mathcal{X} \subset \mathbb{Z}^n$ s.t. we have for each $\lambda \in \mathbb{N}$

$$P_\lambda \subset \mathcal{X}_\lambda. \tag{2.316}$$

In this subsection, we will assume that SKE is of constant depth d over $\mathbb{Z}_q$ for a prime number $q = q(\lambda) > d$. Let $m \in \mathsf{poly}(\lambda)$ be s.t. each ciphertext output by Enc is a vector in $\mathbb{Z}_q^m$ and denote by $\mathsf{Enc}_{\mathsf{off}}$ the offline algorithm of Enc. Additionally, we assume that $\mathbb{Z}_q$ is of width B, i.e., there is some $\varepsilon_{\mathsf{width}} \in \mathsf{negl}(\lambda)$ s.t. we have for each message $x \in \mathcal{X}$

$$\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)}} [||\mathsf{ct}||_\infty > B] \leq \varepsilon_{\mathsf{width}}. \tag{2.317}$$

Our goal in this subsection is to prove Theorem 75, which can be seen as an extension of the attack of Theorem 77 to finite fields. For this end, we will construct an integer SKE scheme $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \mathsf{Dec}')$ for the message space $\mathcal{X}'$ from SKE and show—by a statistical game hop and a cryptographic reduction—that SKE' is IND-CPA secure if SKE is IND-CPA secure. Additionally, Dec' will have (almost) the same decryption probability as Dec, and Enc' will be of depth d and width $d! \cdot B$ over $\mathbb{Z}$. Hence, by using the cryptographic reduction, we can adapt the adversary from Algorithm 10 to SKE.

Now, the integer encryption scheme SKE' is derived from SKE as follows:

Algorithm 12 (Integer SKE Scheme)**.** Let SKE be an SKE scheme of depth d over $\mathbb{Z}_q$ whose message space $\mathcal{X}$ is contained in $\mathbb{Z}^n$.

We construct here an SKE scheme SKE' for a restricted message space $\mathcal{X}' \subset \mathcal{X}$ that is given by

$$\mathcal{X}'_\lambda := \left\{ x \in \mathcal{X}_\lambda \;\middle|\; ||x||_\infty^d < \frac{q}{2 \cdot d! \cdot \Gamma_d \cdot \binom{n+d}{d} \cdot B} \right\}. \tag{2.318}$$

We will describe SKE' by four algorithms $\mathsf{Setup}', \mathsf{Enc}'_{\mathsf{off}}, \mathsf{Enc}', \mathsf{Dec}'$ where the offline algorithm $\mathsf{Enc}'_{\mathsf{off}}$ is only added here for conceptual simplicity:

Setup': Setup$'$ works exactly like Setup. On input 1^λ, it outputs $\mathsf{msk}' := \mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$.

$\mathsf{Enc}'_{\mathsf{off}}$: On input a master secret key msk', $\mathsf{Enc}'_{\mathsf{off}}$ samples $r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$. If there is an $x \in P_\lambda$ s.t.

$$||r(x)||_\infty > B, \tag{2.319}$$

then $\mathsf{Enc}'_{\mathsf{off}}$ returns the vector $(0, \ldots, 0) \in (\mathbb{Z}[X])^m$ of zero-polynomials. Otherwise, it scales $r(X)$ by $d!$ modulo q and interprets the result as a polynomial map in $(\mathbb{Z}[X])^m$. I.e.,

$$r'(X) := (r(X) \cdot d! \bmod q) \in (\mathbb{Z}[X])^m \tag{2.320}$$

where we interpret $r'(X)$ as a collection of integer polynomials whose coefficients lie in $\left[-\frac{q-1}{2}, \frac{q-1}{2}\right]$. Finally, $\mathsf{Enc}'_{\mathsf{off}}$ outputs $r(X) \in (\mathbb{Z}[X_1, \dots X_n]^{\leq d})^m$.

Enc': On input a master secret key msk' and a message $x \in \mathcal{X}'_\lambda$, Enc' samples m integer polynomials $r' \leftarrow \mathsf{Enc}'_{\mathsf{off}}(\mathsf{msk})$ and outputs

$$\mathsf{ct} := r'(x) \in \mathbb{Z}^m \tag{2.321}$$

as ciphertext.

Dec': On input a master secret key msk' and a ciphertext $\mathsf{ct}' \in \mathbb{Z}^m$, Dec' computes the inverse of $d!$ modulo q and sets

$$\mathsf{ct} := (\mathsf{ct}' \bmod q) \cdot (d!)^{-1} \in \mathbb{Z}_q^m. \tag{2.322}$$

It runs $x \leftarrow \mathsf{Dec}(\mathsf{msk}', \mathsf{ct})$ and outputs x as decrypted message.

We can directly read off the depth of SKE' over $\mathbb{Z}$. Since SKE is of width B over $\mathbb{Z}_q$, we can further easily verify the width of SKE':

Proposition 82. *The encryption algorithm* Enc' *of* SKE' *is of depth d and width $d! \cdot B$ over $\mathbb{Z}$. Concretely, we have for each $(x_\lambda)_\lambda \in \mathcal{X}'_\lambda$*

$$\Pr_{\substack{\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda) \\ \mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x_\lambda)}} \left[||\mathsf{ct}'||_\infty > d! \cdot B(\lambda) \right] \leq \varepsilon_{\mathsf{width}}(\lambda). \tag{2.323}$$

We will not directly prove the security of SKE'. Instead, we will consider a similar SKE scheme $\widetilde{\mathsf{SKE}} = (\mathsf{Setup}', \widetilde{\mathsf{Enc}}, \mathsf{Dec}')$ whose ciphertexts are statistically close to the ciphertexts of SKE'. For $\widetilde{\mathsf{SKE}}$, we can give direct security reductions and easily prove correctness. The difference between SKE' and $\widetilde{\mathsf{SKE}}$ is that the encryption algorithm Enc' of SKE' is of depth d, while the encryption algorithm $\widetilde{\mathsf{Enc}}$ of $\widetilde{\mathsf{SKE}}$ is not of constant depth over $\mathbb{Z}$.

To give a formal definition of $\widetilde{\mathsf{Enc}}$, we will use the following auxiliary function:

Definition 35. Denote by

$$\iota : \mathbb{Z}_q[X] \longrightarrow \mathbb{Z}[X] \tag{2.324}$$

the map that maps polynomials over $\mathbb{Z}_q$ to polynomials over $\mathbb{Z}$ by interpreting each coefficient of a polynomial as an integer in $\left\{-\frac{q-1}{2}, \dots, \frac{q-1}{2}\right\}$. Note that we have for each $f \in \mathbb{Z}_q[X]$

$$\iota(f(X)) \bmod q = f(X). \tag{2.325}$$

I.e., ι is a left inverse to _ mod q. Given a vector of polynomials $r = (r_1, \dots, r_m) \in (\mathbb{Z}_q[X])^m$, we will apply ι entry-wise, i.e., $\iota(r) := (\iota(r_1), \dots, \iota(r_m)) \in (\mathbb{Z}[X])^m$.

Note that ι preserve degrees. In particular, $\iota(v)$ is an integer vector if v is a vector of constant polynomials. Hence, we will—by abuse of notation—also use ι as a mapping $\mathbb{Z}_q^m \to \mathbb{Z}^m$.

Algorithm 13. The alternative secret-key encryption scheme $\widetilde{\mathsf{SKE}} = (\mathsf{Setup}', \widetilde{\mathsf{Enc}}, \mathsf{Dec}')$ for the message space $\mathcal{X}'$ uses the same setup and decryption algorithm as SKE' from Algorithm 12. Its encryption algorithm is given as follows:

Enc': On input a master secret key msk' and a message $x \in \mathcal{X}'_\lambda$, $\widetilde{\mathsf{Enc}}$ samples $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)$, which is an element of $\mathbb{Z}_q^m$. It computes $\mathsf{ct} \cdot d!$ modulo q and interprets the result as an integer vector

$$\widetilde{\mathsf{ct}} := \iota(\mathsf{ct} \cdot d! \bmod q) \in \mathbb{Z}^m \tag{2.326}$$

Finally, it outputs $\widetilde{\mathsf{ct}}$ as encryption of x.

Lemma 83. *We have*

$$\mathsf{pr}^{\mathsf{dec}}_{\widetilde{\mathsf{SKE}}}(\mathsf{Dec}') \geq \mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}). \tag{2.327}$$

If SKE *is selectively IND-CPA secure, then so is* $\widetilde{\mathsf{SKE}}$. *Concretely, for each adversary* $\widetilde{\mathcal{A}}$ *against the selective IND-CPA security of* $\widetilde{\mathsf{SKE}}$ *that submits* N *candidate message pairs to the challenger, there is an adversary* $\mathcal{A}$ *against the selective IND-CPA security of* SKE *that submits* N *candidate message pairs to the challenger and that makes* $\Theta(N \cdot m)$ *additional arithmetic operations over* $\mathbb{Z}_q$ *and* $\mathbb{Z}$ *s.t.*

$$\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\widetilde{\mathsf{SKE}}}(\widetilde{\mathcal{A}}) = \mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A}) \tag{2.328}$$

Proof. Let us first prove correctness: draw $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$ and $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)$ for some $x \in \mathcal{X}'_\lambda$. The ciphertext $\widetilde{\mathsf{ct}} \leftarrow \widetilde{\mathsf{Enc}}(\mathsf{msk}, x)$ is equally distributed as $\iota(\mathsf{ct} \cdot d! \bmod q)$. We now have

$$\begin{aligned} \mathsf{Dec}'(\mathsf{msk}, \widetilde{\mathsf{ct}}) =& \mathsf{Dec}'(\mathsf{msk}, \iota(\mathsf{ct} \cdot d! \bmod q)) && (2.329)\\ =& \mathsf{Dec}\big(\mathsf{msk}, (\iota(\mathsf{ct} \cdot d! \bmod q) \bmod q) \cdot (d!)^{-1}\big) && (2.330)\\ =& \mathsf{Dec}\big(\mathsf{msk}, \mathsf{ct} \cdot d! \cdot (d!)^{-1}\big) && (2.331)\\ =& \mathsf{Dec}(\mathsf{msk}, \mathsf{ct}). && (2.332) \end{aligned}$$

It follows that the distributions $\mathsf{Dec}'(\mathsf{msk}, \widetilde{\mathsf{ct}})$ and $\mathsf{Dec}(\mathsf{msk}, \mathsf{ct})$ are identical. In particular, the decryption probability of Dec' must be at least as large as the decryption probability of Dec. (It can even be larger, since the message space of $\widetilde{\mathsf{SKE}}$ may be smaller than the message space of SKE.)

Let $\widetilde{\mathcal{A}}$ be an adversary for the selective IND-CPA security Game 3 of $\widetilde{\mathsf{SKE}}$. We will prove security by constructing a cryptographic reduction $\mathcal{R}$ that plays the selective IND-CPA security game of SKE with a challenger $\mathcal{C}$ and has black-box access to $\widetilde{\mathcal{A}}$:

Step 1: At the start of the game, $\mathcal{R}$ initiates a game for the security of $\widetilde{\mathsf{SKE}}$ with $\widetilde{\mathcal{A}}$ and receives two lists $(x_i^{(0)})_{i=0}^N$ and $(x_i^{(1)})_{i=0}^N$ of messages in $\mathcal{X}'_\lambda$. $\mathcal{R}$ passes both lists on to $\mathcal{C}$ and receives a list of ciphertexts

$$\mathsf{ct}_i \leftarrow \mathsf{Enc}(\mathsf{msk}, x_i^{(b)}), \qquad i = 1, \ldots, N, \tag{2.333}$$

for an unknown bit $b \leftarrow \{0, 1\}$.

Step 2: $\mathcal{R}$ can now turn the ciphertext ct_i to a ciphertext $\widetilde{\mathsf{ct}}_i$ that is distributed according to $\widetilde{\mathsf{Enc}}(\mathsf{msk}, x_i^{(b)})$ by computing

$$\widetilde{\mathsf{ct}}_i := \iota((\mathsf{ct}_i \cdot d!) \bmod q) \in \mathbb{Z}^m, \qquad i = 1, \ldots, N. \tag{2.334}$$

It passes the list $(\widetilde{\mathsf{ct}}_i)_{i=1}^N$ of new ciphertexts on to $\widetilde{\mathcal{A}}$, which responds with a guess $b' \in \{0, 1\}$ that $\mathcal{R}$ refers to $\mathcal{C}$.

$\widetilde{\mathsf{ct}}_i$ is distributed according to $\widetilde{\mathsf{Enc}}(\mathsf{msk}, x_i^{(b)})$ iff ct_i is distributed according to $\mathsf{Enc}(\mathsf{msk}, x_i^{(b)})$. Hence, the view of $\widetilde{\mathcal{A}}$ when interacting with $\mathcal{R}$ is identical to its view in the security game of SKE. Therefore, the advantage of $\mathcal{R}$ with access to $\widetilde{\mathcal{A}}$ against the security of SKE is identical to $\widetilde{\mathcal{A}}$'s advantage against the security of $\widetilde{\mathsf{SKE}}$. Further, for each ciphertext received by $\mathcal{C}$, $\mathcal{R}$ has to perform m arithmetic operations. Ergo, the claim follows. □

In the following, we will show that the statistical distance between $(\mathsf{msk}', \mathsf{ct}')$ for

$$\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda), \qquad \mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x) \tag{2.335}$$

and $(\mathsf{msk}', \widetilde{\mathsf{ct}})$ for

$$\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda), \qquad \widetilde{\mathsf{ct}} \leftarrow \widetilde{\mathsf{Enc}}(\mathsf{msk}', x) \tag{2.336}$$

is negligible. It then follows that the output distributions of $\mathsf{Dec}'(\mathsf{msk}', \mathsf{ct}')$ and $\mathsf{Dec}'(\mathsf{msk}', \widetilde{\mathsf{ct}})$ are statistically close. Further, the view of an adversary attacking SKE' is statistically close to its view when attacking $\widetilde{\mathsf{SKE}}$. Hence, the security of SKE' follows, too.

Lemma 84. *We have for* SKE

$$\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})}} [\exists x \in P_\lambda : \; ||r(x)||_\infty > B] \leq \binom{n+d}{d} \cdot \varepsilon_{\mathsf{width}}(\lambda). \tag{2.337}$$

Proof. The claim easily follows by a union bound:

$$\begin{aligned}
&\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})}} [\exists x \in P_\lambda : \; ||r(x)||_\infty > B] && (2.338)\\
\leq &\sum_{x \in P_\lambda} \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})}} [||r(x)||_\infty > B] && (2.339)\\
\leq &\sum_{x \in P_\lambda} \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)}} [||\mathsf{ct}||_\infty > B] && (2.340)\\
\leq &\# P_\lambda \cdot \varepsilon_{\mathsf{width}}(\lambda) = \binom{n+d}{d} \cdot \varepsilon_{\mathsf{width}}(\lambda).
\end{aligned}$$

□

Lemma 85. *If $q > d$, we have for* SKE'

$$\Pr_{\substack{\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda) \\ r' \leftarrow \mathsf{Enc}'_{\mathsf{off}}(\mathsf{msk}')}} [\deg r' > 0] \geq \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})}} [\deg r > 0] - \binom{n+d}{d} \cdot \varepsilon_{\mathsf{width}}(\lambda).$$

Proof. Since q is coprime to each number $1, \ldots, d$, we have for $r \in (\mathbb{Z}_q[X])^m$

$$\deg(r) = \deg(\iota(d! \cdot r \bmod q)). \tag{2.341}$$

Because of Lemma 84, the probability that $\mathsf{Enc}'_{\mathsf{off}}$ outputs 0 instead of $\iota(d! \cdot r \bmod q)$ (when sampling $r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}'))$ is lower bounded by $\binom{n+d}{d}\varepsilon_{\mathsf{width}}$. Hence, the claim follows. □

Lemma 86. *Let $f \in \mathbb{Z}_q[X]$ be of degree d and $x \in \mathbb{Z}^n$ s.t.*

$$||\mathrm{coeff}_d(f)||_\infty \cdot ||x||_\infty{}^d \cdot \binom{n+d}{d} < q/2. \tag{2.342}$$

Then, we have the following equality over $\mathbb{Z}$

$$\iota(f)(x) = \iota(f(x)). \tag{2.343}$$

Proof. Set $g := \iota(f)$. Eq. (2.343) is equivalent to

$$g(x) = \iota(g(x) \bmod q). \tag{2.344}$$

Note that $g(x)$ and $\iota(g(x) \bmod q)$ must be equal modulo q and that $\iota(g(x) \bmod q)$ must be an integer in $\{-(q-1)/2, \ldots, (q-1)/2\}$. Hence, it suffices to show that $|g(x)|$ is bounded by $q/2$. Denote by $y \in \mathbb{Z}^{\binom{n+d}{d}}$ the vector that contains each product of $\leq d$ entries of x as element. We have

$$\begin{aligned} |g(x)| \leq & ||\mathrm{coeff}_d(g)||_\infty \cdot ||y||_1 \leq ||\mathrm{coeff}_d(g)||_\infty \cdot ||y||_\infty \cdot \binom{n+d}{d} && (2.345) \\ = & ||\mathrm{coeff}_d(f)||_\infty \cdot ||x||_\infty{}^d \cdot \binom{n+d}{d} < q/2, && (2.346) \end{aligned}$$

where the last inequality follows from Eq. (2.342). □

Lemma 87. *Let $f \in \mathbb{Z}_q[X]$ be of degree d. Assume that we have for each $x \in P_\lambda$*

$$|f(x)| \leq B. \tag{2.347}$$

Then, it follows

$$||d! \cdot \mathrm{coeff}_d(f)||_\infty \leq d! \cdot \Gamma_d \cdot B. \tag{2.348}$$

Proof. Set $L = \binom{n+d}{d}$ and let $\alpha^{(n)}(1), \ldots, \alpha^{(n)}(L)$ be the enumeration of all points in P_λ from Convention 2.

Let $g := \iota(f)$ be the integer version of f. We have seen in Section 2.2 that it holds

$$\mathrm{coeff}_d(g) = V(n,d)^{-1} \cdot \begin{pmatrix} g(\alpha^{(n)}(1)) \\ \vdots \\ g(\alpha^{(n)}(L)) \end{pmatrix}. \tag{2.349}$$

Now, $V(n,d)^{-1}$ is not an integer matrix. However, we have proven in Theorem 54 that $W(n,d) = d! \cdot V(n,d)^{-1}$ is integer. Hence, we have

$$\begin{aligned} & d! \cdot \mathrm{coeff}_d(f) = (d! \cdot \mathrm{coeff}_d(g)) \bmod q && (2.350) \\ = & W(n,d) \cdot \begin{pmatrix} g(\alpha^{(n)}(1)) \\ \vdots \\ g(\alpha^{(n)}(L)) \end{pmatrix} \bmod q = W(n,d) \cdot \begin{pmatrix} f(\alpha^{(n)}(1)) \\ \vdots \\ f(\alpha^{(n)}(L)) \end{pmatrix}. && (2.351) \end{aligned}$$

Now, the infinity norm of $(f(\alpha^{(n)}(1)),\ldots,f(\alpha^{(n)}(L)))$ is bounded by B. For the infinity norm of $W(n,d)$ we have $||W(n,d)||_\infty = \left|\left|d!\cdot V(n,d)^{-1}\right|\right|_\infty \le d!\cdot\Gamma_d$. Hence, we get

$$||d!\cdot \mathrm{coeff}_d(f)||_\infty \le \left|\left| W(n,d)\cdot \begin{pmatrix} f(\alpha^{(n)}(1)) \\ \vdots \\ f(\alpha^{(n)}(L)) \end{pmatrix} \right|\right|_\infty \tag{2.352}$$

$$\le ||W(n,d)||_\infty \cdot \left|\left|(f(\alpha^{(n)}(1)),\ldots,f(\alpha^{(n)}(L)))\right|\right|_\infty \le d!\cdot\Gamma_d\cdot B. \qquad \square$$

Lemma 88. *Now, for $x\in\mathcal{X}_\lambda$, let $(\mathsf{msk}',\mathsf{ct}')$ and $(\mathsf{msk}',\widetilde{\mathsf{ct}})$ be the distributions that first sample $\mathsf{msk}'\leftarrow\mathsf{Setup}'(1^\lambda)$, and then $\mathsf{ct}'\leftarrow\mathsf{Enc}'(\mathsf{msk}',x)$ and $\widetilde{\mathsf{ct}}\leftarrow\widetilde{\mathsf{Enc}}(\mathsf{msk}',x)$, respectively. We have*

$$\Delta((\mathsf{msk}',\mathsf{ct}'),(\mathsf{msk}',\widetilde{\mathsf{ct}})) \le \binom{n+d}{d}\cdot\varepsilon_{\mathsf{width}}(\lambda). \tag{2.353}$$

Proof. We will first describe the algorithms $\mathsf{Enc}'(\mathsf{msk}',x)$ and $\widetilde{\mathsf{Enc}}(\mathsf{msk}',x)$ as deterministic functions $f_r'(x)$ and $\widetilde{f}_r(x)$ of $r\leftarrow\mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}')$ and x. For this end, set

$$R_\lambda := \{r\in\mathbb{Z}_q[X_1,\ldots,X_n] \mid \forall x\in P_\lambda : ||r(x)||_\infty \le B\}. \tag{2.354}$$

If $\mathsf{Enc}_{\mathsf{off}}'(\mathsf{msk}')$ samples r from $\mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}')$, it will output $\iota(d!\cdot r(X))$, if $r\in R_\lambda$, and 0, otherwise. The output of $\mathsf{Enc}_{\mathsf{off}}'(\mathsf{msk}')$ will be evaluated by $\mathsf{Enc}'(\mathsf{msk}',x)$ on x. Hence, we have

$$\mathsf{Enc}'(\mathsf{msk}',x) = f_r'(x) := \begin{cases} \iota(d!\cdot r)(x), & \text{if } r\in R_\lambda, \\ 0, & \text{otherwise.} \end{cases} \tag{2.355}$$

If Enc samples $r\leftarrow\mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}')$, it evaluates r at x and passes the result $r(x)$ on to $\widetilde{\mathsf{Enc}}$. $\widetilde{\mathsf{Enc}}(\mathsf{msk}',x)$ will scale $r(x)$ by $d!$ and interpret the product as an integer vector, which it will output as ciphertext. Hence, we have

$$\widetilde{\mathsf{Enc}}(\mathsf{msk}',x) = \widetilde{f}_r(x) := \iota(d!\cdot r(x)). \tag{2.356}$$

We claim that the terms in Eqs. (2.355) and (2.356) are equal whenever $r\in P_\lambda$, i.e., we have for each $x\in\mathcal{X}_\lambda'$

$$f_r'(x) = \iota(d!\cdot r)(x) = \iota(d!\cdot r(x)) = \widetilde{f}_r(x). \tag{2.357}$$

Indeed, this follows from Lemma 86, since we have for each $i\in[m]$

$$||\mathrm{coeff}_d(d!\cdot r_i)||_\infty \cdot ||x||_\infty^d \cdot \binom{n+d}{d} \tag{2.358}$$

$$=||d!\cdot\mathrm{coeff}_d(r_i)||_\infty \cdot ||x||_\infty^d \cdot \binom{n+d}{d} \tag{2.359}$$

$$\overset{Lemma\ 87}{\le} d!\cdot\Gamma_d\cdot B\cdot ||x||_\infty^d \cdot \binom{n+d}{d} \overset{Eq.\ (2.318)}{<} q/2. \tag{2.360}$$

Now, fix some master secret key msk' and consider the statistical distance between $\mathsf{Enc}'(\mathsf{msk}', x)$ and $\widetilde{\mathsf{Enc}}(\mathsf{msk}', x)$

$$\begin{aligned}
&\Delta(\mathsf{Enc}'(\mathsf{msk}', x), \widetilde{\mathsf{Enc}}(\mathsf{msk}', x)) && (2.361)\\
=&\frac{1}{2}\sum_{\mathsf{ct}'}\left|\Pr\left[\mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x)\right] - \Pr\left[\mathsf{ct}' \leftarrow \widetilde{\mathsf{Enc}}(\mathsf{msk}', x)\right]\right| && (2.362)\\
=&\frac{1}{2}\sum_{\mathsf{ct}'}\left|\Pr_{r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}')}[\mathsf{ct}' = f'_r(x)] - \Pr_{r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}')}\left[\mathsf{ct}' = \widetilde{f}_r(x)\right]\right|. && (2.363)
\end{aligned}$$

Let $\mathbb{1}_{a=b}$ be the characteristic function that is 1, if $a = b$, and 0, otherwise. We have

$$\begin{aligned}
&\frac{1}{2}\sum_{\mathsf{ct}'}\left|\Pr_{r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}')}[\mathsf{ct}' = f'_r(x)] - \Pr_{r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}')}\left[\mathsf{ct}' = \widetilde{f}_r(x)\right]\right| && (2.364)\\
=&\frac{1}{2}\sum_{r}\Pr[r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}')] \cdot \left|\mathbb{1}_{\mathsf{ct}'=f'_r(x)} - \mathbb{1}_{\mathsf{ct}'=\widetilde{f}_r(x)}\right| && (2.365)\\
=&\frac{1}{2}\sum_{r}\Pr[r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}')] \cdot \sum_{\mathsf{ct}'}\left|\mathbb{1}_{\mathsf{ct}'=f'_r(x)} - \mathbb{1}_{\mathsf{ct}'=\widetilde{f}_r(x)}\right| && (2.366)\\
=&\frac{1}{2}\sum_{r}\Pr[r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}')] \cdot 2 \cdot \mathbb{1}_{f'_r(x)\neq\widetilde{f}_r(x)} && (2.367)\\
=&\Pr_{r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}')}\left[f'_r(x) \neq \widetilde{f}_r(x)\right] && (2.368)\\
\overset{Eq.\ (2.357)}{\leq}&\Pr_{r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}')}\left[\exists x \in P_\lambda :\ ||r(x)||_\infty > B\right] && (2.369)
\end{aligned}$$

Finally, we can bound the statistical distance $\Delta((\mathsf{msk}', \mathsf{Enc}'(\mathsf{msk}', x)), (\mathsf{msk}', \widetilde{\mathsf{Enc}}(\mathsf{msk}', x)))$, by

$$\begin{aligned}
&\Delta((\mathsf{msk}', \mathsf{Enc}'(\mathsf{msk}', x)), (\mathsf{msk}', \widetilde{\mathsf{Enc}}(\mathsf{msk}', x))) && (2.370)\\
=&\sum_{\mathsf{msk}'}\Pr[\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda)] \cdot \Delta(\mathsf{Enc}'(\mathsf{msk}', x), \widetilde{\mathsf{Enc}}(\mathsf{msk}', x)) && (2.371)\\
\leq&\sum_{\mathsf{msk}'}\Pr[\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda)] && (2.372)\\
&\cdot \Pr_{r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}')}\left[\exists x \in P_\lambda :\ ||r(x)||_\infty > B\right] && (2.373)\\
=&\Pr_{\substack{\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda)\\ r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}')}}\left[\exists x \in P_\lambda :\ ||r(x)||_\infty > B\right] && (2.374)\\
\overset{Lemma\ 84}{\leq}&\binom{n+d}{d} \cdot \varepsilon_{\mathsf{width}}(\lambda).
\end{aligned}$$

□

Lemma 89. *The SKE scheme* SKE' *from Algorithm 12 is correct and selectively IND-CPA secure if* SKE *is correct and selectively IND-CPA secure, respectively. Concretely, we have*

$$\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') \geq \mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}) - \binom{n+d}{d} \cdot \varepsilon_{\mathsf{width}}, \qquad (2.375)$$

and for each adversary $\mathcal{A}'$ against the selective IND-CPA security of SKE' *that makes N encryption queries there is an adversary $\mathcal{A}$ against the selective IND-CPA security of* SKE *that makes the same number of encryption queries. The time complexity of $\mathcal{A}$ equals the time complexity of $\mathcal{A}'$ plus $\Theta(mN)$ additional arithmetic operations over $\mathbb{Z}$ and $\mathbb{Z}_q$. We have*

$$\mathsf{adv}_{\mathsf{SKE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}) \geq \mathsf{adv}_{\mathsf{SKE}'}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}') - N \cdot \binom{n+d}{d} \cdot \varepsilon_{\mathsf{width}}. \tag{2.376}$$

Proof. Let $(x_\lambda)_\lambda \in \mathcal{X}'$. Because of Lemma 88, the statistical distance between $(\mathsf{msk}', \mathsf{Enc}'(\mathsf{msk}', x_\lambda))$ and $(\mathsf{msk}', \widetilde{\mathsf{Enc}}(\mathsf{msk}', x_\lambda))$ is bounded by $\binom{n+d}{d} \cdot \varepsilon_{\mathsf{width}}(\lambda)$, for $\mathsf{msk}' \leftarrow \mathsf{Setup}(1^\lambda)$. Hence, the distance between $\mathsf{Dec}'(\mathsf{msk}', \mathsf{Enc}'(\mathsf{msk}', x_\lambda))$ and $\mathsf{Dec}'(\mathsf{msk}', \widetilde{\mathsf{Enc}}(\mathsf{msk}', x_\lambda))$ is bounded by $\binom{n+d}{d} \cdot \varepsilon_{\mathsf{width}}(\lambda)$, too. For the decryption probability, we therefore have

$$\begin{aligned}
\mathsf{pr}_{\mathsf{SKE}'}^{\mathsf{dec}}(\mathsf{Dec}') &= \min_{x \in \mathcal{X}'_\lambda} \Pr\left[\mathsf{Dec}'(\mathsf{msk}', \mathsf{Enc}'(\mathsf{msk}', x)) = x\right] \\
&\geq \min_{x \in \mathcal{X}'_\lambda} \Pr\left[\mathsf{Dec}'(\mathsf{msk}', \widetilde{\mathsf{Enc}}(\mathsf{msk}', x)) = x\right] - \binom{n+d}{d} \cdot \varepsilon_{\mathsf{width}}(\lambda) \\
&= \mathsf{pr}_{\widetilde{\mathsf{SKE}}}^{\mathsf{dec}}(\mathsf{Dec}') - \binom{n+d}{d} \cdot \varepsilon_{\mathsf{width}}(\lambda) \\
&\overset{\textit{Lemma } 83}{\geq} \mathsf{pr}_{\mathsf{SKE}}^{\mathsf{dec}}(\mathsf{Dec}) - \binom{n+d}{d} \cdot \varepsilon_{\mathsf{width}}(\lambda).
\end{aligned}$$

Now, let $\mathcal{A}'$ be an adversary that plays the selective IND-CPA security Game 3 of SKE'. Let $(x_i^{(0)})_{i=1}^N$, $(x_i^{(1)})_{i=1}^N$ be the two lists of messages submitted by $\mathcal{A}'$, and let $(\mathsf{ct}'_i)_{i=1}^N$ be the ciphertexts returned by the challenger. For $i \in [N]$, set

$$\widetilde{\mathsf{ct}}_i := \widetilde{\mathsf{Enc}}(\mathsf{msk}', x_i^{(b)}) \tag{2.377}$$

where $\mathsf{msk}' \leftarrow \mathsf{Setup}(1^\lambda)$ and $b \leftarrow \{0,1\}$ are the master secret key and the bit drawn by the challenger. Lemmas 50 and 88 imply

$$\Delta((\mathsf{ct}'_1, \ldots, \mathsf{ct}'_N), (\widetilde{\mathsf{ct}}_1, \ldots, \widetilde{\mathsf{ct}}_N)) \leq N \cdot \binom{n+d}{d} \cdot \varepsilon_{\mathsf{width}}(\lambda).$$

Hence, the statistical distance between the view of $\mathcal{A}'$ in the security game of SKE' and in the security game of $\widetilde{\mathsf{SKE}}$ is bounded. Lemma 83 now yields the existence of an adversary $\mathcal{A}$ against SKE s.t.

$$\begin{aligned}
\mathsf{adv}_{\mathsf{SKE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}) &= \mathsf{adv}_{\widetilde{\mathsf{SKE}}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}') \qquad (2.378) \\
&\geq \mathsf{adv}_{\mathsf{SKE}'}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}') - N \cdot \binom{n+d}{d} \cdot \varepsilon_{\mathsf{width}}(\lambda). \qquad \square
\end{aligned}$$

We will now prove Theorem 75, with which we started Section 2.4.

Proof Theorem 75. Let $\mathsf{SKE} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{Dec})$ be an SKE for messages

$$\{0, 1, \ldots, 2d\} \subset \mathcal{X}. \tag{2.379}$$

Further, let Enc be of constant depth d over $\mathbb{Z}_q$. Assume that we have

$$\forall (x_\lambda)_\lambda \in \mathcal{X} : \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda)}} [||\mathsf{ct}||_\infty > B] \leq \varepsilon_{\mathsf{width}}(\lambda), \tag{2.380}$$

$$2 \cdot (d+1)! \cdot (2d)^d \cdot \Gamma_d \cdot B < q. \tag{2.381}$$

Let $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \mathsf{Dec}')$ be the SKE construction from Algorithm 12 for SKE with message space

$$\mathcal{X}' = \left\{ x \in \mathcal{X} \;\middle|\; |x|^d < \frac{q}{2 \cdot (d+1)! \cdot \Gamma_d \cdot B} \right\}. \tag{2.382}$$

Because of Lemmas 52, 85 and 89 and Proposition 82, we have:

1. $0, \ldots, 2d \in \mathcal{X}'$.
2. SKE' is of depth d over $\mathbb{Z}_q$.
3. Set $B' := d! \cdot B$. We have for each $(x_\lambda)_\lambda \in \mathcal{X}'$

$$\Pr_{\substack{\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda) \\ \mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x_\lambda)}} [||\mathsf{ct}'||_\infty > B'] \leq \varepsilon_{\mathsf{width}}(\lambda). \tag{2.383}$$

4. We have

$$\begin{aligned} \Pr_{\substack{\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda) \\ r' \leftarrow \mathsf{Enc}'_{\mathsf{off}}(\mathsf{msk}')}} [\deg r' > 0] \geq & \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})}} [\deg r > 0] - (d+1)\varepsilon_{\mathsf{width}} \\ \geq & \mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}) - (d+1)\varepsilon_{\mathsf{width}}. \end{aligned}$$

5. For each adversary $\mathcal{A}'$ against SKE' that makes N encryption queries there is an adversary $\mathcal{A}$ against SKE that makes the same number of encryption queries and $\Theta(mN)$ more arithmetical operations over $\mathbb{Z}$ and $\mathbb{Z}_q$. Additionally, we have

$$\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A}) \geq \mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}'}(\mathcal{A}') - (d+1) \cdot N \cdot \varepsilon_{\mathsf{width}}. \tag{2.384}$$

Set

$$p := \mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}) - (d+1)\varepsilon_{\mathsf{width}}, \tag{2.385}$$

$$N' := \left\lceil \frac{16 \cdot \Gamma_{2d} \cdot m \cdot B'^2 \cdot (1 + 2 \cdot \Gamma_d \cdot B')^d}{p - \varepsilon_{\mathsf{width}}} \right\rceil \tag{2.386}$$

$$= \left\lceil \frac{16 \cdot \Gamma_{2d} \cdot m \cdot B'^2 \cdot (1 + 2 \cdot \Gamma_d \cdot B')^d}{\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}) - (d+2)\varepsilon_{\mathsf{width}}} \right\rceil. \tag{2.387}$$

According to Theorem 77, the adversary $\mathcal{A}'$ from Algorithm 10 instantiated with d, B' and N' makes $3N$ ciphertext queries and $\Theta(mN)$ arithmetical operations over $\mathbb{Z}$ for $N = N'^3$. It has an advantage of

$$\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}'}(\mathcal{A}') \tag{2.388}$$

$$\geq \frac{p - \varepsilon_{\mathsf{width}}}{4dm} - 16\exp(-2N') - 3N\varepsilon_{\mathsf{width}} \tag{2.389}$$

$$= \frac{\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}) - (d+2)\varepsilon_{\mathsf{width}}}{4dm} - 16\exp(-2N') - 3N\varepsilon_{\mathsf{width}}. \tag{2.390}$$

Hence, there exists an adversary $\mathcal{A}$ against the IND-CPA security of SKE that has the same asymptotic runtime complexity, makes the same number of encryption queries and whose advantage is lower bounded by

$$
\begin{aligned}
&\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}}(\mathcal{A}) && (2.391)\\
\geq &\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}'}(\mathcal{A}') - N \cdot (d+1) \cdot \varepsilon_{\mathsf{width}} && (2.392)\\
= &\frac{\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}}(\mathsf{Dec}) - (d+2)\varepsilon_{\mathsf{width}}}{4dm} - 16\exp(-2N') - (d+4) \cdot N \cdot \varepsilon_{\mathsf{width}}. && \square
\end{aligned}
$$

2.5 Lower Bounds for FE

We will conclude this chapter by proving lower bounds for lattice-based FE schemes. In Section 2.5.1, we will give a template method for proving such lower bounds for advanced functionalities. In Section 2.5.2, we will show—without a lot of effort—that this method implies lower bounds for *function-hiding* FE. Finally, in Section 2.5.4 we will use the same method again to give lower bounds for the security of *compact* FE, however, this time we will need to put in more work and rely on a theorem about homogeneity among ciphertexts that we will prove in Section 2.5.3.

2.5.1 A Template Method

The idea of this section is to give a general strategy for proving lower bounds in the lattice-based FE framework that we sketched in Definition 30. For this end, we will use the efficient adversaries of Section 2.4 on the IND-CPA security of SKE schemes of constant depth and small width over prime fields $\mathbb{Z}_q$. In Theorem 75, we showed that any encryption scheme that encrypts numbers by applying random polynomials and produces short ciphertexts bounded by some $B \ll q/2$ cannot be secure (against adversaries of time complexity $\mathsf{poly}(B)$) if they are noticeably correct. Our strategy is to extend this adversary to lattice-based FE schemes. Remember that we agreed in Definition 30 to call an FE scheme $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ *lattice-based*, if the following things hold:

- Enc is of constant depth d_1 over $\mathbb{Z}_q$ (and each ciphertext output by Enc lies in $\mathbb{Z}_q^m$),
- Each secret key output by KeyGen is a polynomial in $\mathbb{Z}_q[C_1, \ldots, C_m]$ of constant degree d_2.
- For each ciphertext $\mathsf{ct} \in \mathbb{Z}_q^m$ and each secret key $\mathsf{sk} \in \mathbb{Z}_q[C]$ we have

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = 0 \iff |\mathsf{sk}(\mathsf{ct})| < B \tag{2.393}$$

  for some noise bound $B \ll q/2$.

Now, Enc is of constant depth d_1 over $\mathbb{Z}_q$, so this requirement of Theorem 75 is fulfilled. However, Theorem 75 additionally requires Enc to be of small width, which will not be the case, in general. To solve this problem, we will use secret keys $\mathsf{sk} \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f)$ of functions f that evaluate to zero on the messages we will consider here. Concretely, let $x \in \mathcal{X}$ be some message and let $f \in \mathcal{F}$ be a function that vanishes on x, i.e., $f(x) = 0$. Since FE is correct, we have with overwhelming probability for $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$, $\mathsf{sk} \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f)$ and $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)$

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = f(x) = 0. \tag{2.394}$$

Since FE is lattice-based, this implies

$$|\mathsf{sk}(\mathsf{ct})| < B \tag{2.395}$$

with overwhelming probability over the randomness of $\mathsf{msk}, \mathsf{ct}$ and sk.

This gives rise to the following strategy: let $\mathcal{X} = \mathbb{Z}_p^n$ be the message space of our scheme and let $\mathcal{X}' \subset \mathbb{Z}_p^n$ be a special subspace (for example, $\mathcal{X}' = \mathbb{Z}_p \times \{0\} \times \ldots \times \{0\}$). Let $f_1, \ldots, f_Q \in \mathcal{F}$ be a list of functions that map each element of $\mathcal{X}'$ to zero and let $f_* \in \mathcal{F}$ be a function that is injective on $\mathcal{X}'$ (for example, f_* maps each vector to its first coordinate). Let $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$ and sk_* be the secret keys of $f_1, \ldots, f_Q$ and f_*, respectively.

Now, to encrypt $x \in \mathcal{X}'$, first sample $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)$ and then compute a new ciphertext

$$\mathsf{ct}' := (\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_Q(\mathsf{ct})). \tag{2.396}$$

Since each f_i evaluates to zero on x, each entry $\mathsf{sk}_i(\mathsf{ct})$ of ct' must be bounded by B with overwhelming probability. Further, the values $\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_Q(\mathsf{ct})$ do not leak substantial about $x \in \mathcal{X}'$ according to the IND-CPA Game 3. Hence, if FE is secure in the sense of Game 3, then ct' must hide which message $x \in \mathcal{X}'$ it encrypts. Concretely, there is a reduction that reduces the security of FE to the security of this new encryption procedure. However, is this encryption algorithm, let us call it Enc', of constant depth over $\mathbb{Z}_q$? It turns out it is: Enc itself is of depth d_1 and Enc' applies Q polynomials of degree d_2 on top of Enc (note that the polynomials $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$ are independent of x). Hence, the depth of Enc' is $d_1 \cdot d_2$.

Enc' gives rise to a partial SKE scheme $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \_)$ with message space $\mathcal{X}'$ on which we can apply Theorem 75. It follows that there cannot exist any (computationally unbounded) decryption algorithm Dec' s.t. the SKE $(\mathsf{Setup}', \mathsf{Enc}', \mathsf{Dec}')$ has a non-negligible decryption advantage (since, otherwise, the adversary from Theorem 75 would have a non-negligible advantage against the IND-CPA security of SKE' which implies an adversary with non-negligible advantage against the IND-CPA security of FE). In Sections 2.5.2 and 2.5.4, we will show that such a decryptor, or rather extractor, does indeed exist if FE is function-hiding or compact, respectively. This raises a contradiction to the required IND-CPA security and correctness of FE and yields our lower bound.

Let us formalize the above argument. We start by giving a formal description of the partial SKE scheme $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \_)$ that can be derived from FE. Our description will follow the generalization given in [TÜ23].

Algorithm 14. Let q be a prime and $p < q$ be a modulus for the message space. Let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be a functional encryption scheme with message space $\mathcal{X} = \mathbb{Z}_p^n$ and value space $\mathcal{Y} = \mathbb{Z}_p$ s.t. FE is lattice-based with encryption depth $d_1 \in O(1)$, decryption depth $d_2 \in O(1)$ and noise bound $B < q/2$ over $\mathbb{Z}_q$.

We construct a partial SKE $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \_)$ with (yet unspecified) parameters Q and $M \leq (p-1)/2$. The message space of SKE' is $\mathcal{X}' := \{0, \ldots, M\}$. The algorithms of SKE' are given by:

$\mathsf{Setup}'_{\mathsf{Pre}}$: There is a preceding setup algorithm that on input 1^λ chooses functions $f_1, \ldots, f_Q, f_* \in \mathcal{F}_\lambda$. Further, it chooses an affine linear map

$$\nu : \mathbb{Z}_p \longrightarrow \mathbb{Z}_p^n, \tag{2.397}$$

s.t. we have for all $x \in \mathcal{X}'_\lambda = \{0, \ldots, M(\lambda)\}$

$$\forall i \in [Q] : f_i(\nu(x)) = 0, \tag{2.398}$$

$$f_*(\nu(x)) = x. \tag{2.399}$$

It outputs $(f_1, \ldots, f_Q, f_*, \nu)$.

Setup' : On input 1^λ, Setup' runs $(f_1, \ldots, f_Q, f_*, \nu) \leftarrow \mathsf{Setup}'_{\mathsf{Pre}}(1^\lambda)$.
Then, Setup' computes $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$, $\mathsf{sk}_i \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_i)$ for $i \in [Q]$ and $\mathsf{sk}_* \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_*)$. It outputs the new master secret key

$$\mathsf{msk}' := (\mathsf{msk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_Q, \mathsf{sk}_*, \nu). \tag{2.400}$$

Enc' : On input $\mathsf{msk}' = (\mathsf{msk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_Q, \mathsf{sk}_*, \nu)$ and a message $x \in \{0, \ldots, M\}$, Enc' runs $\mathsf{ct}_x \leftarrow \mathsf{Enc}(\mathsf{msk}, \nu(x))$ and outputs the new ciphertext

$$\mathsf{ct}'_x := (\mathsf{sk}_1(\mathsf{ct}_x), \ldots, \mathsf{sk}_Q(\mathsf{ct}_x)). \tag{2.401}$$

Remark 8. We will assume that the linear embedding

$$\nu : \mathbb{Z}_p \longrightarrow \mathbb{Z}_p^n, \tag{2.402}$$

specified by $\mathsf{Setup}'_{\mathsf{Pre}}$ is of degree 1 over the *integers*. In other words, we interpret $\mathbb{Z}_p$ as $\{0, \ldots, p-1\} \subset \mathbb{Z}$ or $\{\lceil -p/2 \rceil, \ldots, \lfloor (p-1)/2 \rfloor\} \subset \mathbb{Z}$ and assume that ν can be computed over $\mathbb{Z}$ *without* arithmetic reductions modulo p. We require this, because we want to compose ν with polynomial maps over $\mathbb{Z}_q$. If we compose a polynomial $f \in \mathbb{Z}_p[X]$ with a polynomial $g \in \mathbb{Z}_q[Y]$, then the composition $g \circ f$ does not need to be a polynomial over $\mathbb{Z}_q$. However, if f is a polynomial $f \in \mathbb{Z}[X]$, then the composition $g \circ f$ lies canonically in $\mathbb{Z}_q[X]$.

The linear embeddings ν that we study in Sections 2.5.2 and 2.5.4 will be of very simple nature. In particular, they are of degree 1 over the integers.

In the Lemmas 90 to 92, we will show that $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \_)$ is of constant depth and small width over $\mathbb{Z}_q$. Further, we will show that SKE' is secure if FE is secure.

Lemma 90. *In the scheme* $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \_)$ *from Algorithm 14,* Enc' *is of depth* $d_1 \cdot d_2$.

Proof. Since FE has encryption depth d_1, there is an algorithm $\mathsf{Enc}_{\mathsf{off}}$ that on input msk outputs m polynomials $r_1, \ldots, r_m \in \mathbb{Z}_q[X_1, \ldots, X_n]$ of degree $\leq d_1$ s.t. $\mathsf{Enc}(\mathsf{msk}, x)$ is equally distributed as $(r_1(x), \ldots, r_m(x))$ for each $x \in \mathbb{Z}_p^n$. We define an offline algorithm $\mathsf{Enc}'_{\mathsf{off}}$ for Enc' as follows:

$\mathsf{Enc}'_{\mathsf{off}}$: On input $\mathsf{msk}' = (\mathsf{msk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_Q, \mathsf{sk}_*, \nu)$, $\mathsf{Enc}'_{\mathsf{off}}$ first samples

$$r = (r_1, \ldots, r_m) \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk}). \tag{2.403}$$

For each $i \in [m]$, it computes

$$r'_i(X) := \mathsf{sk}_i \circ r \circ \nu = \mathsf{sk}_i(r_1(\nu(X)), \ldots, r_m(\nu(X))) \in \mathbb{Z}_q[X_1, \ldots, X_n].$$

Finally, it outputs $r' = (r'_1(X), \ldots, r'_Q(X))$.

The degree of each r'_i is bounded by $d_1 \cdot d_2 \cdot 1$, since each sk_i is a polynomial in $\mathbb{Z}_q[C]$ of degree $\leq d_2$, r is a tuple of m polynomials of degree $\leq d_1$, and ν is an affine linear function, i.e. a degree-1 polynomial, over the integers

We claim that for each $x \in \mathcal{X}'_\lambda$ and msk' we have the equality of random variables

$$\mathsf{ct}' = r'(x) \tag{2.404}$$

for $r' \leftarrow \mathsf{Enc}'_{\mathsf{off}}(\mathsf{msk}')$ and $\mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x)$. Indeed, Enc' generates ciphertexts by sampling $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, \nu(x))$ and outputting $\mathsf{ct}' = (\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_Q(\mathsf{ct}))$. Since $\mathsf{Enc}_{\mathsf{off}}$ is the offline algorithm of Enc, we have for $r \leftarrow \mathsf{Enc}_{\mathsf{off}}(\mathsf{msk})$ the following equalities of distributions

$$\begin{aligned}\mathsf{ct}' =& (\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_Q(\mathsf{ct})) && (2.405)\\ =& (\mathsf{sk}_1(r(\nu(x))), \ldots, \mathsf{sk}_Q(r(\nu(x)))) && (2.406)\\ =& (r'_1(x), \ldots, r'_Q(x)) = r'(x) && (2.407)\end{aligned}$$

(where $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$ are fixed by msk'). □

Lemma 91. *If* FE *is correct, then* Enc' *is of width* B. *Concretely, we have for each* $(x_\lambda)_\lambda \in \mathcal{X}'$

$$\Pr_{\substack{\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda)\\ \mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x_\lambda)}} [||\mathsf{ct}'||_\infty > B] \leq Q(\lambda) \cdot (1 - \mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})). \tag{2.408}$$

Proof. Let $x \in \mathcal{X}'_\lambda$ and let $j \in [Q(\lambda)]$. Sample $\mathsf{msk}' = (\mathsf{msk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_Q, \mathsf{sk}_*, \nu) \leftarrow \mathsf{Setup}'(1^\lambda)$ and let f_j be the function evaluated by sk_j. Since FE is lattice-based with noise-bound B and since $f_j(\nu(x)) = 0$, we have

$$\begin{aligned}&\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)\\ \mathsf{sk}_j \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_j)\\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, \nu(x))}} [|\mathsf{sk}_j(\mathsf{ct})| > B] && (2.409)\\ \leq &\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)\\ \mathsf{sk}_j \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_j)\\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, \nu(x))}} [\mathsf{Dec}(\mathsf{sk}_j, \mathsf{ct}) \neq f_j(\nu(x))] \leq 1 - \mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}). && (2.410)\end{aligned}$$

The claimed inequality now follows by a union bound

$$\begin{aligned}&\Pr_{\substack{\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda)\\ \mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x)}} [||\mathsf{ct}'||_\infty > B] && (2.411)\\ =&\Pr_{\substack{(f_1, \ldots, f_Q, f_*, \nu) \leftarrow \mathsf{Setup}'_{\mathsf{Pre}}(1^\lambda)\\ \mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)\\ \forall j:\ \mathsf{sk}_j \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_j)\\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, \nu(x))}} [\exists j \in [Q] :\ |\mathsf{sk}_j(\mathsf{ct})| > B] && (2.412)\\ \leq&\sum_{j=1}^{Q} \Pr_{\substack{(f_1, \ldots, f_Q, f_*, \nu) \leftarrow \mathsf{Setup}'_{\mathsf{Pre}}(1^\lambda)\\ \mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)\\ \mathsf{sk}_j \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_j)\\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, \nu(x))}} [|\mathsf{sk}_j(\mathsf{ct})| > B] && (2.413)\\ \leq&\sum_{j=1}^{Q} (1 - \mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})) = Q - Q \cdot \mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}). && \square\end{aligned}$$

Lemma 92. *If* FE *is selectively IND-CPA secure, then* SKE' *is selectively IND-CPA secure.*

To be precise, for each adversary $\mathcal{A}'$ *on the selective IND-CPA security of* SKE' *that makes* N *encryption queries there is an adversary* $\mathcal{A}$ *on the selective IND-CPA security of* FE *that makes* N *encryption and* Q *function queries. In addition to the arithmetic operations performed by* $\mathcal{A}'$ *and* $\mathsf{Setup}'_{\mathsf{Pre}}$, $\mathcal{A}$ *performs* $O(nN)$ *operations over* $\mathbb{Z}_p$ *and* $O\big(m^{d_2} \cdot QN\big)$ *operations over* $\mathbb{Z}_q$.

For the advantage of $\mathcal{A}$*, we have*

$$\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{FE}}(\mathcal{A}) = \mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}'}(\mathcal{A}'). \tag{2.414}$$

Proof. Let $\mathcal{A}'$ be an adversary against the selective IND-CPA security of SKE'. We will construct a reduction $\mathcal{R}$ that plays the selective IND-CPA security Game 3 of FE with a challenger $\mathcal{C}$ while imitating a challenger for the selective IND-CPA security Game 3 of SKE' for $\mathcal{A}'$.

$\mathcal{R}$ proceeds in the following steps:

Step 1: $\mathcal{R}$ computes

$$(f_1, \dots, f_Q, f_*, \nu) \leftarrow \mathsf{Setup}'_{\mathsf{Pre}}(1^\lambda). \tag{2.415}$$

Step 2: $\mathcal{R}$ starts $\mathcal{A}'$ and receives two lists $(x'^{(0)}_i)_{i=1}^N$, $(x'^{(1)}_i)_{i=1}^N$ of messages in $\mathcal{X}'_\lambda$. For each $i \in [N]$ and $\beta \in \{0,1\}$, it sets

$$x_i^{(\beta)} := \nu(x'^{(\beta)}_i) \in \mathcal{X}_\lambda. \tag{2.416}$$

Step 3: $\mathcal{R}$ submits the lists $(x_i^{(0)})_{i=1}^N$, $(x_i^{(1)})_{i=1}^N$ and $(f_j)_{j=1}^Q$ to $\mathcal{C}$, and receives in response ciphertexts

$$\mathsf{ct}_i \leftarrow \mathsf{Enc}(\mathsf{msk}, x_i^{(b)}) \ \text{ for } i \in [N] \tag{2.417}$$

and secret keys

$$\mathsf{sk}_j \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_j) \ \text{ for } j \in [Q], \tag{2.418}$$

for an unknown bit $b \leftarrow \{0,1\}$ and an unknown master secret key $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$ sampled by $\mathcal{C}$.

Step 4: For each $i \in [N]$, $\mathcal{R}$ computes a new ciphertext by

$$\mathsf{ct}'_i := (\mathsf{sk}_1(\mathsf{ct}_i), \dots, \mathsf{sk}_Q(\mathsf{ct}_i)) \in \mathbb{Z}_q^Q. \tag{2.419}$$

It sends the list $(\mathsf{ct}'_i)_{i=1}^N$ to $\mathcal{A}'$.

Step 5: $\mathcal{A}'$ replies with a guess b' that $\mathcal{R}$ passes on to $\mathcal{C}$.

The view of $\mathcal{A}'$ in the interaction with $\mathcal{R}$ is identical to its view in Game 3 of SKE'. Furthermore, $\mathcal{R}$ wins exactly iff $\mathcal{A}'$ wins. This is, because we have for all $i \in [N]$ and $j \in [Q]$

$$f_j(x_i^0) = f_j(\nu(x'^0_i)) = 0 = f_j(\nu(x'^1_i)) = f_j(x_i^1). \tag{2.420}$$

In other words, $\mathcal{R}$ is a valid adversary in the selective IND-CPA Game 3 of FE. In particular, $\mathcal{R}$ does not submit to $\mathcal{C}$ any combination of functions and message pairs that would help it to win trivially. It follows that the advantage of $\mathcal{R}$ in the security game of FE is equal to the advantage of $\mathcal{A}'$ in the security game of SKE'.

Finally, the time complexity of $\mathcal{R}$ is dominated by the time complexities of $\mathsf{Setup}'_{\mathsf{Pre}}$ and $\mathcal{A}'$. Additionally, in Step 2, $\mathcal{R}$ has to apply a degree-1 function to each message submitted by $\mathcal{A}'$, which amounts to $O(nN)$ arithmetic operations over $\mathbb{Z}_p$. In Step 4, $\mathcal{R}$ has to apply Q degree-d_2 polynomials to each ciphertext received by $\mathcal{C}$, which adds $O\Big(\binom{m+d_2}{d_2} \cdot NQ\Big)$ arithmetical operations over $\mathbb{Z}_q$. □

With the Lemmas 90 and 92 and Theorem 75, we can now prove that FE must be insecure if there exists a successful decryptor Dec' for SKE'.

Theorem 93. *Let* FE *be a lattice-based and correct FE scheme of encryption depth* d_1*, decryption depth* d_2 *and noise-bound* B*. Let* $\mathcal{X} = \mathbb{Z}_p^n$ *be the message space and* $\mathcal{Y} = \mathbb{Z}_p$ *the value space of* FE*, and assume that each ciphertext of* FE *is a vector in* $\mathbb{Z}_q^m$ *for* $q > p > 2$*, where* q *is prime.*

Let $M, Q \in \mathbb{N}$ *be parameters. Set* $d := d_1 \cdot d_2$*,* $\varepsilon_{\mathsf{width}} := Q \cdot (1 - \mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}))$*, and assume that the following inequalities hold:*

1. $2d \leq M < p/2$*,*
2. $2 \cdot (d+1)! \cdot (2d)! \cdot \Gamma_d \cdot B < q$*.*

Let $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \_)$ *be the partial SKE scheme from Algorithm 14 that is constructed from* FE *with message space* $\mathcal{X}' = \{0, \ldots, M\}$*, and let* Dec' *be a decryption algorithm for* SKE'*. Let* $N' \in \mathbb{N}$ *s.t.*

$$N' \geq \frac{16 \cdot (d!)^2 \cdot \Gamma_{2d} \cdot Q \cdot B^2 \cdot (1 + 2 \cdot d! \cdot \Gamma_d \cdot B)^d}{\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') - (d+2)\varepsilon_{\mathsf{width}}} \tag{2.421}$$

and set $N := N'^3$*.*

There exists an adversary $\mathcal{A}$ *against the selective IND-CPA security of* FE *that makes* $3N$ *encryption and* Q *function queries. In addition to the arithmetic operations made by* $\mathsf{Setup}'_{\mathsf{Pre}}$*,* $\mathcal{A}$ *makes*

$$O(m^{d_2} \cdot QN + nN) \tag{2.422}$$

arithmetic operations over $\mathbb{Z}_p, \mathbb{Z}_q$ *and* $\mathbb{Z}$*. In Game 3,* $\mathcal{A}$ *has an advantage of*

$$\geq \frac{\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') - (d+2) \cdot \varepsilon_{\mathsf{width}}}{4dQ} - 16\exp(-2N') - (d+4) \cdot N \cdot \varepsilon_{\mathsf{width}}.$$

Proof. Let $\mathsf{Setup}', \mathsf{Enc}'$ be the algorithms from Algorithm 14 for some $\mathsf{Setup}'_{\mathsf{Pre}}$, and let Dec' be some decryption algorithm. By abuse of notation, we will set $\mathsf{SKE}' := (\mathsf{Setup}', \mathsf{Enc}', \mathsf{Dec}')$.

Lemma 90 states that SKE' is of depth $d = d_1 \cdot d_2$, and Lemma 91 states that we have

$$\Pr_{\substack{\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda) \\ \mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x)}} [||\mathsf{ct}'||_\infty > B] \leq \varepsilon_{\mathsf{width}} = Q \cdot (1 - \mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})) \tag{2.423}$$

for each $x \in \mathcal{X}'_\lambda$.

Theorem 75 now gives us an adversary $\mathcal{A}'$ against the selective IND-CPA security of SKE' that makes $3N$ encryption queries and $\Theta(QN)$ arithmetic operations over $\mathbb{Z}_q$ and $\mathbb{Z}$.

For the advantage of $\mathcal{A}'$, we have

$$\begin{aligned}\mathsf{adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{SKE}'}(\mathcal{A}') \geq & \frac{\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') - (d+2)\cdot\varepsilon_{\mathsf{width}}}{4dQ} && (2.424)\\ & - 16\exp(-2N') - (d+4)\cdot N\cdot\varepsilon_{\mathsf{width}}. && (2.425)\end{aligned}$$

With Lemma 92, there is an adversary $\mathcal{A}$ against the selective IND-CPA security of FE that makes $3N$ encryption and Q function queries.

In addition to the operations performed by $\mathcal{A}'$ and $\mathsf{Setup}'_{\mathsf{Pre}}$, $\mathcal{A}$ makes $O(nN)$ arithmetic operations over $\mathbb{Z}_p$ and $O(m^{d_2}\cdot QN)$ operations over $\mathbb{Z}_q$. □

Remark 9. Note again that the adversary $\mathcal{A}$ from Theorem 93 performs arithmetic operations over $\mathbb{Z}$. For the lower bounds of Sections 2.5.2 and 2.5.4, we can approximate the adversary $\mathcal{A}$ by an algorithm that only performs bit operations. Because the time complexity of $\mathsf{Setup}'_{\mathsf{Pre}}$ will be omittable and n will be smaller than m^{d_2} for the results Theorems 94 and 108, $\mathcal{A}$ will perform in both cases

$$O(m^{d_2}\cdot QN) \tag{2.426}$$

arithmetic operations over $\mathbb{Z}_p, \mathbb{Z}_q$ and $\mathbb{Z}$. Now, each integer operation of $\mathcal{A}$ will involve numbers, which are bounded by Nq^2. Hence, in the world of Turing machines, $\mathcal{A}$'s behaviour can be approximated by

$$O\big((\log(q)^2 + \log(N)^2)\cdot m^{d_2}\cdot QN\big) \tag{2.427}$$

bit operations. For $N' \in O\Big(\frac{QB^{2+d}}{\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}')}\Big)$, this yields a bit complexity of

$$O\bigg(\Big(\log(q)^2 + \log(Q)^2 + \log(\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}'))^2\Big)\cdot m^{d_2}\cdot\frac{Q^4B^{6+3d}}{\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}')^3}\bigg).$$

Asymptotically, Theorem 93 states that, if there exists algorithms $\mathsf{Setup}'_{\mathsf{Pre}}$ and Dec' s.t. $(\mathsf{Setup}', \mathsf{Enc}', \mathsf{Dec}')$ has a non-negligible decryption advantage, then there exists a $\mathsf{poly}(B+\lambda)$-time attacker on FE (assuming that FE is correct). We will close this section with two examples of simple inner-product FE schemes where the template method outlined here will fail and succeed, respectively.

Example 1 (Simple Inner-Product Encryption)**.** We will give here a very simple FE scheme where the function space $\mathcal{F} : \mathbb{Z}_p^n \to \mathbb{Z}$ is the space of linear functions (we will omit arithmetic reductions modulo p).

Setup: On input 1^λ, Setup outputs as master secret key a random matrix $S \leftarrow \mathbb{Z}_q^{n\times n}$.

KeyGen: On input $S \in \mathbb{Z}_q^{n\times n}$ and $y \in \mathbb{Z}_p^n$, KeyGen interprets y as $\mathbb{Z}_q$-vector and outputs

$$\mathsf{sk} := \begin{pmatrix} -Sy \\ y \end{pmatrix} \in \mathbb{Z}_q^{2n}. \tag{2.428}$$

Enc: On input $x \in \mathbb{Z}_p^n$ and $S \in \mathbb{Z}_q^{n \times n}$, Enc interprets x as $\mathbb{Z}_q$-vector. It samples $a \leftarrow \mathbb{Z}_q^n$ and $e \leftarrow \chi^n$, where χ is a noise distribution over $\mathbb{Z}$ that is bounded by $q/(4n \cdot p^2)$. Enc outputs the ciphertext

$$\mathsf{ct} := \begin{pmatrix} a \\ Sa + e + \left\lceil \frac{q}{n \cdot p^2} \right\rceil \cdot x \end{pmatrix} \in \mathbb{Z}_q^{2n}. \tag{2.429}$$

Dec : On input $\mathsf{sk}, \mathsf{ct} \in \mathbb{Z}_q^{2n}$, Dec computes

$$z := \mathsf{sk}^T \cdot \mathsf{ct} \bmod q \in \mathbb{Z}_q. \tag{2.430}$$

Then, it rounds z to $\mathbb{Z}_{n \cdot p^2}$, i.e. it finds a value $a \in \left\{0, \ldots, n \cdot p^2 - 1\right\}$ s.t.

$$\left| z - \left\lceil \frac{q}{n \cdot p^2} \right\rceil \cdot a \bmod q \right| \tag{2.431}$$

is minimal. Finally, it outputs $a \bmod p$ as value.

We omit here the proof of correctness. Note that—if x and y are orthogonal—we have

$$\mathsf{sk}^T \mathsf{ct} = -y^T Sa + y^T Sa + y^T e + \left\lceil \frac{q}{n \cdot p^2} \right\rceil \cdot y^T x = y^T e, \tag{2.432}$$

which is smaller than $B := q/(2p)$ modulo q. Hence, the scheme $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ is lattice-based of encryption and decryption depth 1 and noise-bound B in the sense of Definition 30 (where we interpret the secret keys as linear functions).

We can now create an SKE SKE' from FE. According to Algorithm 14, we only need to specify an algorithm $\mathsf{Setup}'_{\mathsf{Pre}}$ that on input 1^λ outputs a list of n functions ($Q = n - 1$) and one linear embedding. We give the following algorithm for $\mathsf{Setup}'_{\mathsf{Pre}}$:

$\mathsf{Setup}'_{\mathsf{Pre}}$: For $i \in [n]$, let $e_i \in \mathbb{Z}_p^n$ be the unit vector that is one at position i. On input 1^λ, $\mathsf{Setup}_{\mathsf{Pre}}$ sets

$$y_i := e_i \tag{2.433}$$

for $i = 1, \ldots, n-1$ and

$$y_* := e_n. \tag{2.434}$$

It lets $\nu : \mathbb{Z}_p \to \mathbb{Z}_p^n$ be the linear map that maps x to $x \cdot e_n$. It outputs $(y_1, \ldots, y_{n-1}, y_*, \nu)$.

The message space of SKE' is $\mathcal{X}' = \mathbb{Z}_p$. Note that each vector of $y_1, \ldots, y_{n-1}$ is orthogonal to the image of ν, while we have for y_*

$$y_*^T \cdot \nu(x) = e_n^T \cdot e_n \cdot x = x. \tag{2.435}$$

If S is the master secret key of FE, then the master secret key of SKE' is given by

$$\mathsf{msk}' = (S, \mathsf{sk}_1, \ldots, \mathsf{sk}_{n-1}, \mathsf{sk}_*, \nu) \tag{2.436}$$

where $\mathsf{sk}_i = (-Sy_i, y_i)$ is the secret key for y_i.

Denote by $E \in \mathbb{Z}_q^{(n-1)\times 2n}$ the matrix whose i-th row is sk_i^T. On input S and $x \in \mathbb{Z}_p$, Enc' outputs

$$\begin{aligned}
\mathsf{ct}' = E \cdot \mathsf{Enc}(S, \nu(x)) &= \begin{pmatrix} -y_1^T S & y_1^T \\ \vdots & \vdots \\ -y_{n-1}^T S & y_{n-1}^T \end{pmatrix} \cdot \begin{pmatrix} a \\ Sa + e + \left\lceil \frac{q}{n \cdot p^2} \right\rceil \cdot \nu(x) \end{pmatrix} \\
&= \begin{pmatrix} y_1^T e + \left\lceil \frac{q}{n \cdot p^2} \right\rceil \cdot y_1^T \nu(x) \\ \vdots \\ y_{n-1}^T e + \left\lceil \frac{q}{n \cdot p^2} \right\rceil \cdot y_{n-1}^T \nu(x) \end{pmatrix} = \begin{pmatrix} y_1^T e \\ \vdots \\ y_{n-1}^T e \end{pmatrix}.
\end{aligned}$$

Note that ct' is devoid of any information about x. There is no decryption algorithm Dec' that has a non-negligible decryption advantage for SKE'. In fact, this is because FE can be proven to be secure (under a suitable LWE assumption).

We see in this case that our template method for showing lower bounds has no success, because the left-over noise $y_i^T e$ contains no information about the encrypted message x.

Example 2 (Simple Function-Hiding Inner Product Encryption)**.** The example FE given in Example 1 is not function-hiding. We will attempt here to fix this and give a new (candidate) function-hiding inner-product FE scheme:

Setup: On input 1^λ, Setup outputs as master secret key a random matrix $S \leftarrow \mathbb{Z}_q^{n\times n}$.

KeyGen: On input $S \in \mathbb{Z}_q^{n\times n}$ and $y \in \mathbb{Z}_p^n$, KeyGen interprets y as $\mathbb{Z}_q$-vector. It samples some small noise $f \leftarrow \chi^n$ and outputs

$$\mathsf{sk} := S^{-T} \cdot \left(\left\lceil \sqrt{\frac{q}{n \cdot p^2}} \right\rceil \cdot y + f \right). \tag{2.437}$$

Enc: On input $x \in \mathbb{Z}_p^n$ and $S \in \mathbb{Z}_q^{n\times n}$, Enc interprets x as $\mathbb{Z}_q$-vector. It samples $e \leftarrow \chi^n$ and outputs

$$\mathsf{ct} := S \cdot \left(\left\lceil \sqrt{\frac{q}{n \cdot p^2}} \right\rceil \cdot x + e \right). \tag{2.438}$$

Dec : On input $\mathsf{sk}, \mathsf{ct} \in \mathbb{Z}_q^{2n}$, Dec computes

$$z := \mathsf{sk}^T \cdot \mathsf{ct} \bmod q \in \mathbb{Z}_q. \tag{2.439}$$

Then, it rounds z to $\mathbb{Z}_{n\cdot p^2}$, i.e. it finds a value $a \in \left\{0, \ldots, n \cdot p^2 - 1\right\}$ s.t.

$$\left| z - \left\lceil \sqrt{\frac{q}{n \cdot p^2}} \right\rceil^2 \cdot a \bmod q \right| \tag{2.440}$$

is minimal. Finally, it outputs $a \bmod p$ as value.

Again, we omit the proof of correctness. Now, the leftover noise of FE for orthogonal x, y is given by

$$\begin{aligned}\mathsf{sk}^T \cdot \mathsf{ct} &= \left(\left\lceil\sqrt{\frac{q}{n \cdot p^2}}\right\rceil \cdot y + f\right)^T \cdot S^{-1} \cdot S \cdot \left(\left\lceil\sqrt{\frac{q}{n \cdot p^2}}\right\rceil \cdot x + e\right) && (2.441)\\ &= \left\lceil\sqrt{\frac{q}{n \cdot p^2}}\right\rceil^2 \cdot y^T x + \left\lceil\sqrt{\frac{q}{n \cdot p^2}}\right\rceil \cdot (y^T e + f^T x) + f^T e && (2.442)\\ &= \left\lceil\sqrt{\frac{q}{n \cdot p^2}}\right\rceil \cdot (y^T e + f^T x) + f^T e. && (2.443)\end{aligned}$$

Note that this is problematic. If we want to distinguish two messages, let us say $x_0 = (0, \dots, 0)$ and $x_1 = (p-1, \dots, p-1)$, we can simply set y to zero and try to measure the magnitude difference of $f^T e$ and $\left\lceil\sqrt{\frac{q}{n \cdot p^2}}\right\rceil \cdot f^T x_1 + f^T e$. Indeed, the proposed scheme is insecure. In the next section, we will show that this leakage by left-over noise is inherent for the task of lattice-based function-hiding FE.

2.5.2 Function-Hiding Functional Encryption

We will prove here the first of our two lower bounds for functional encryption schemes:

Theorem 94 (Lower Bounds for Function-Hiding FE)**.** *Let FE be an FE scheme for the functionality of linear functions $\mathcal{F} : \mathbb{Z}_p^n \to \mathbb{Z}_p$. Further, let FE be lattice-based of encryption depth d_1, decryption depth d_2 and noise-bound $B > 0$. Set $d := d_1 \cdot d_2$ and assume that each ciphertext of FE lies in $\mathbb{Z}_q^m$ and that we have*

$$2d < p/2 \qquad \text{and} \qquad 2 \cdot (d+1)! \cdot (2d)! \cdot \Gamma_d \cdot B < q. \tag{2.444}$$

For each $Q = Q(\lambda)$ there is a function $\varepsilon_{\mathsf{fh}}$ s.t. the following two things hold:

1. *Let $N' \in \mathbb{N}$ with*

$$N' \geq \frac{16 \cdot (d!)^2 \cdot \Gamma_{2d} \cdot Q \cdot B^2 \cdot (1 + 2 \cdot d! \cdot \Gamma_d \cdot B)^d}{(1 - \binom{m+d_2}{d_2}/Q - \varepsilon_{\mathsf{fh}}) \cdot \mathsf{adv}_{\mathsf{FE}}^{\mathsf{dec}}(\mathsf{Dec}) - (d+2)\varepsilon_{\mathsf{width}}} \tag{2.445}$$

   where $\varepsilon_{\mathsf{width}} = Q \cdot (1 - \mathsf{pr}_{\mathsf{FE}}^{\mathsf{dec}}(\mathsf{Dec}))$, and set $N := N'^3$.

   There is an adversary $\mathcal{A}^{\mathsf{ind\text{-}cpa}}$ on the selective IND-CPA security of FE that makes $3N$ encryption, Q function queries and $O(m^{d_2} \cdot Q \cdot N)$ arithmetic operations over $\mathbb{Z}_p, \mathbb{Z}_q$ and $\mathbb{Z}$. In Game 3, $\mathcal{A}^{\mathsf{ind\text{-}cpa}}$ has an advantage of

$$\begin{aligned}\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}^{\mathsf{ind\text{-}cpa}}) \geq &\frac{(1 - \binom{m+d_2}{d_2}/Q - \varepsilon_{\mathsf{fh}}) \cdot \mathsf{adv}_{\mathsf{FE}}^{\mathsf{dec}}(\mathsf{Dec}) - (d+2) \cdot \varepsilon_{\mathsf{width}}}{4dQ}\\ &- 16\exp(-2N') - (d+4) \cdot N \cdot \varepsilon_{\mathsf{width}}.\end{aligned}$$

2. *There is an adversary $\mathcal{A}^{\mathsf{fh}}$ on the selective function-hiding security of FE that makes $\Theta(Q \cdot m^{2d_2})$ arithmetic operations over $\mathbb{Z}_q$ and $\mathbb{Z}_p$ and whose advantage is at least $\varepsilon_{\mathsf{fh}}$.*

Corollary 95. *Let* FE *be an FE scheme for linear functions over* $\mathbb{Z}_p^n$ *that is lattice-based of constant encryption and decryption depth and noise-bound* B*. Assume that we have*

$$m \in \mathsf{poly}(\lambda), \qquad p \in \omega(1), \qquad q \in \omega(B) \qquad \text{and} \qquad q \in 2^{\mathsf{poly}(\lambda)} \tag{2.446}$$

If FE *is correct and selectively function-hiding secure against PPT adversaries, then it is not selectively IND-CPA secure against adversaries of time-complexity* $\mathsf{poly}(\lambda + B)$*.*

Proof. Set $Q := m^{d_2} \cdot \lambda$. Since m and Q are polynomial and since FE is function-hiding secure, we have

$$\varepsilon_{\mathsf{fh}} \in \mathsf{negl}(\lambda). \tag{2.447}$$

Now,

$$N' := \left\lceil \frac{16 \cdot (d!)^2 \cdot \Gamma_{2d} \cdot Q \cdot B^2 \cdot (1 + 2 \cdot d! \cdot \Gamma_d \cdot B)^d}{(1 - \binom{m+d_2}{d_2}/Q - \varepsilon_{\mathsf{fh}}) \cdot \mathsf{adv}_{\mathsf{FE}}^{\mathsf{dec}}(\mathsf{Dec}) - (d+2)\varepsilon_{\mathsf{width}}} \right\rceil \in \Theta\big(Q \cdot B^{2+d}\big)$$

lies in $\mathsf{poly}(\lambda + B)$. In particular, the runtime of $\mathcal{A}^{\mathsf{ind\text{-}cpa}}$ from Theorem 94 is polynomial in λ and B, and we have for its advantage

$$\begin{aligned} \mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}^{\mathsf{ind\text{-}cpa}}) \geq & \frac{(1 - \binom{m+d_2}{d_2}/Q - \varepsilon_{\mathsf{fh}}) \cdot \mathsf{adv}_{\mathsf{FE}}^{\mathsf{dec}}(\mathsf{Dec}) - (d+2) \cdot \varepsilon_{\mathsf{width}}}{4dQ} \\ & - 16\exp(-2N') - (d+4) \cdot N \cdot \varepsilon_{\mathsf{width}} \\ \geq & \frac{1}{8dQ} - \mathsf{negl}(\lambda), \end{aligned}$$

since $\mathsf{adv}_{\mathsf{FE}}^{\mathsf{dec}}(\mathsf{Dec})$ is overwhelming. □

We will prove Theorem 94 by using the strategy outlined in Section 2.5.1. For this end, we will specify the following $\mathsf{Setup}'_{\mathsf{Pre}}$ and Dec' algorithms:

Algorithm 15. Let $Q \in \mathbb{N}$. Let $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \mathsf{Dec}')$ be the secret-key encryption scheme for $\mathcal{X}' = \mathbb{Z}_p$ that uses the Setup' and Enc' algorithms from Algorithm 14 that are based on the following preceding setup algorithm:

$\mathsf{Setup}'_{\mathsf{Pre}}$: On input 1^λ, $\mathsf{Setup}'_{\mathsf{Pre}}$ denotes by $f_1, \dots, f_Q$ the zero function of $\mathcal{F}$. By $f_* : \mathbb{Z}_p^n \to \mathbb{Z}_p$ it denotes the linear function that maps each vector to its first output, and by $\nu : \mathbb{Z}_p \to \mathbb{Z}_p^n$ it denotes the linear map that sends x to $(x, 0, \dots, 0)$. It outputs

$$(f_1, \dots, f_Q, f_*, \nu). \tag{2.448}$$

The decryption algorithm Dec' of SKE' is given as follows:

Dec': On input the master secret key

$$\mathsf{msk}' = (\mathsf{msk}, \mathsf{sk}_1, \dots, \mathsf{sk}_Q, \mathsf{sk}_*, \nu) \tag{2.449}$$

(where sk_i is the secret key of f_i for $i \in \{1, \ldots, Q, *\}$) and a ciphertext $\mathsf{ct}' = (\mathsf{ct}'_1, \ldots, \mathsf{ct}'_Q) \in \mathbb{Z}_q^Q$, Dec' checks if sk_* lies in $\operatorname{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1, \ldots, \mathsf{sk}_Q\}$. If so, it computes $\alpha_1, \ldots, \alpha_Q \in \mathbb{Z}_q$ s.t.

$$\mathsf{sk}_*(C) = \alpha_1 \cdot \mathsf{sk}_1(C) + \ldots + \alpha_Q \cdot \mathsf{sk}_Q(C). \tag{2.450}$$

It computes $\mathsf{sk}_*(\mathsf{ct}) := \alpha_1 \cdot \mathsf{ct}'_1 + \ldots + \alpha_Q \cdot \mathsf{ct}'_q$ and uses it to compute $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$. (Note that this works, since FE is lattice-based and the output of $\mathsf{Dec}(\mathsf{sk}_*, \mathsf{ct})$ only depends on $\mathsf{sk}_*(\mathsf{ct})$.) Otherwise, it samples $x' \leftarrow \mathbb{Z}_p$ uniformly at random and outputs x'.

Because of Theorem 93, it suffices to show that Dec' has a non-negligible advantage at decryption if Q is large enough. For this end, we will show the following simple lemma about learning vector spaces:

Lemma 96. *Let k be a field, $m \in \mathbb{N}$ and let $\mathcal{D}$ be a memoryless discrete distribution over k^m. For $\ell \in \mathbb{N}$, we have*

$$\Pr_{v_1,\ldots,v_\ell \leftarrow \mathcal{D}}[v_\ell \in \operatorname{span}_k\{v_1, \ldots, v_{\ell-1}\}] \geq 1 - \frac{m}{\ell}. \tag{2.451}$$

Proof. Let $\ell > m$ and fix $v_1, \ldots, v_\ell \in k^m$. Denote by S^ℓ the group of permutations of the set $[\ell]$ and by $T \subset S^\ell$ the subgroup of order ℓ which is generated by the cyclic rotation $(123\ldots\ell)$. Since each v_i is an m-dimensional vector we have

$$\begin{aligned} \ell - m \leq & \#\{j \in [\ell] \mid v_j \in \operatorname{span}_k\{v_i \mid i \in [\ell], i \neq j\}\} && (2.452)\\ = & \#\big\{\tau \in T \;\big|\; v_{\tau(\ell)} \in \operatorname{span}_k\big\{v_{\tau(1)}, \ldots, v_{\tau(\ell-1)}\big\}\big\}. && (2.453) \end{aligned}$$

Hence, for each fixed choice $v_1, \ldots, v_\ell \in k^m$ we have

$$\Pr_{\tau \leftarrow T}\big[v_{\tau(\ell)} \in \operatorname{span}_k\big\{v_{\tau(1)}, \ldots, v_{\tau(\ell-1)}\big\}\big] \geq \frac{\ell - m}{\ell}. \tag{2.454}$$

Since the vectors $v_1, \ldots, v_\ell$ are identically and independently distributed, we have

$$\begin{aligned} & \Pr_{v_1,\ldots,v_\ell \leftarrow \mathcal{D}}[v_\ell \in \operatorname{span}_k\{v_1, \ldots, v_{\ell-1}\}] && (2.455)\\ = & \Pr_{\substack{v_1,\ldots,v_\ell \leftarrow \mathcal{D}\\ \tau \leftarrow T}}\big[v_{\tau(\ell)} \in \operatorname{span}_k\big\{v_{\tau(1)}, \ldots, v_{\tau(m-1)}\big\}\big]. && (2.456) \end{aligned}$$

Combining our observations, we get

$$\begin{aligned} & \Pr_{v_1,\ldots,v_\ell \leftarrow \mathcal{D}}[v_\ell \in \operatorname{span}_k\{v_1, \ldots, v_{\ell-1}\}]\\ = & \Pr_{\substack{v_1,\ldots,v_\ell \leftarrow \mathcal{D}\\ \tau \leftarrow T}}\big[v_{\tau(\ell)} \in \operatorname{span}_k\big\{v_{\tau(1)}, \ldots, v_{\tau(\ell-1)}\big\}\big]\\ = & \sum_{v_1,\ldots,v_\ell \in k^m} \Pr\big[(v_1, \ldots, v_\ell) \leftarrow \mathcal{D}^\ell\big] \cdot \Pr_{\tau \leftarrow T}\big[v_{\tau(\ell)} \in \operatorname{span}_k\big\{v_{\tau(1)}, \ldots, v_{\tau(m-1)}\big\}\big]\\ \geq & \sum_{v_1,\ldots,v_\ell \in k^m} \Pr\big[(v_1, \ldots, v_\ell) \leftarrow \mathcal{D}^\ell\big] \cdot \frac{\ell - m}{\ell} = \frac{\ell - m}{\ell} = 1 - \frac{m}{\ell}. \end{aligned}$$

□

Lemma 97. *Let* $(f_{*,\lambda})_\lambda = f_* : \mathbb{Z}_p^n \to \mathbb{Z}_p$ *be the linear function that maps each vector to its first coordinate. Let*

$$\varepsilon_{\mathsf{fh}} = 1 - \frac{1}{Q+1}\binom{m+d_2}{d_2} - \Pr_{\substack{\mathsf{msk}\leftarrow\mathsf{Setup}(1^\lambda)\\ \mathsf{sk}_1,\ldots,\mathsf{sk}_Q\leftarrow\mathsf{KeyGen}(\mathsf{msk},0)\\ \mathsf{sk}_*\leftarrow\mathsf{KeyGen}(\mathsf{msk},f_{*,\lambda})}}\Big[\mathsf{sk}_* \in \mathrm{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1,\ldots,\mathsf{sk}_Q\}\Big].$$

There is an adversary on the selective function-hiding security of FE *that makes* $\Theta(Q \cdot m^{2d_2})$ *arithmetic operations over* $\mathbb{Z}_q$ *and* $\mathbb{Z}_p$ *and whose advantage is at least* $\varepsilon_{\mathsf{fh}}$.

Proof. Consider the following adversary $\mathcal{A}^{\mathsf{fh}}$ on the selective function-hiding security of FE that plays the selective function-hiding security Game 4 with a challenger:

Step 1: At the start of the game, $\mathcal{A}^{\mathsf{fh}}$ sets for $j \in [Q]$

$$f_j^{(0)} := f_j^{(1)} := 0 \tag{2.457}$$

and

$$f_{Q+1}^{(0)} := 0, \qquad f_{Q+1}^{(1)} := f_{*,\lambda}. \tag{2.458}$$

Step 2: It submits the lists $(f_j^{(0)})_{j=1}^{Q+1}$, $(f_j^{(1)})_{j=1}^{Q+1}$ of functions to the challenger and receives a list of secret keys $\mathsf{sk}_1, \ldots, \mathsf{sk}_{Q+1}$.

Step 3: If

$$\mathsf{sk}_{Q+1} \in \mathrm{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1,\ldots,\mathsf{sk}_Q\}, \tag{2.459}$$

then $\mathcal{A}^{\mathsf{fh}}$ sends 0 to the challenger. Otherwise, it sends 1.

We have for the advantage of $\mathcal{A}^{\mathsf{fh}}$

$$\begin{aligned}
&\mathsf{adv}_{\mathsf{FE}}^{\mathsf{FH}}(\mathcal{A}^{\mathsf{fh}}) && (2.460)\\
={}& \Pr_{\substack{\mathsf{msk}\leftarrow\mathsf{Setup}(1^\lambda)\\ \mathsf{sk}_1,\ldots,\mathsf{sk}_Q\leftarrow\mathsf{KeyGen}(\mathsf{msk},0)\\ \mathsf{sk}_{Q+1}\leftarrow\mathsf{KeyGen}(\mathsf{msk},0)}}\Big[\mathsf{sk}_{Q+1} \in \mathrm{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1,\ldots,\mathsf{sk}_Q\}\Big] && (2.461)\\
&+ \Pr_{\substack{\mathsf{msk}\leftarrow\mathsf{Setup}(1^\lambda)\\ \mathsf{sk}_1,\ldots,\mathsf{sk}_Q\leftarrow\mathsf{KeyGen}(\mathsf{msk},0)\\ \mathsf{sk}_{Q+1}\leftarrow\mathsf{KeyGen}(\mathsf{msk},f_{*,\lambda})}}\Big[\mathsf{sk}_{Q+1} \notin \mathrm{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1,\ldots,\mathsf{sk}_Q\}\Big] - 1 && (2.462)\\
={}& \Pr_{\substack{\mathsf{msk}\leftarrow\mathsf{Setup}(1^\lambda)\\ \mathsf{sk}_1,\ldots,\mathsf{sk}_Q\leftarrow\mathsf{KeyGen}(\mathsf{msk},0)\\ \mathsf{sk}_{Q+1}\leftarrow\mathsf{KeyGen}(\mathsf{msk},0)}}\Big[\mathsf{sk}_{Q+1} \in \mathrm{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1,\ldots,\mathsf{sk}_Q\}\Big] && (2.463)\\
&- \Pr_{\substack{\mathsf{msk}\leftarrow\mathsf{Setup}(1^\lambda)\\ \mathsf{sk}_1,\ldots,\mathsf{sk}_Q\leftarrow\mathsf{KeyGen}(\mathsf{msk},0)\\ \mathsf{sk}_{Q+1}\leftarrow\mathsf{KeyGen}(\mathsf{msk},f_{*,\lambda})}}\Big[\mathsf{sk}_{Q+1} \in \mathrm{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1,\ldots,\mathsf{sk}_Q\}\Big]. && (2.464)
\end{aligned}$$

Each secret key is polynomial of degree d_2 over m variables. As a vector space, $\mathbb{Z}_q[C_1, \ldots, C_m]^{\leq d_2}$ has dimension $\binom{m+d_2}{d_2}$. Hence, Lemma 96 implies

$$\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{sk}_1,\ldots,\mathsf{sk}_Q \leftarrow \mathsf{KeyGen}(\mathsf{msk},0) \\ \mathsf{sk}_{Q+1} \leftarrow \mathsf{KeyGen}(\mathsf{msk},0)}} \Big[\mathsf{sk}_{Q+1} \in \mathrm{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1, \ldots, \mathsf{sk}_Q\}\Big] \geq 1 - \frac{1}{Q+1} \cdot \binom{m+d_2}{d_2}. \tag{2.465}$$

Hence, we have

$$\mathsf{adv}^{\mathsf{FH}}_{\mathsf{FE}}(\mathcal{A}^{\mathsf{fh}}) \geq 1 - \frac{\binom{m+d_2}{d_2}}{Q+1} - \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{sk}_1,\ldots,\mathsf{sk}_Q \leftarrow \mathsf{KeyGen}(\mathsf{msk},0) \\ \mathsf{sk}_{Q+1} \leftarrow \mathsf{KeyGen}(\mathsf{msk},f_{*,\lambda})}} \Big[\mathsf{sk}_{Q+1} \in \mathrm{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1, \ldots, \mathsf{sk}_Q\}\Big].$$

The time complexity of $\mathcal{A}^{\mathsf{fh}}$ is dominated by eliminating a $Q \times \binom{m+d_2}{d_2}$-matrix which costs $\Theta(Q \cdot m^{2d_2})$ arithmetic operations over $\mathbb{Z}_q$. □

Lemma 98. *Let $\varepsilon_{\mathsf{fh}}$ be the function from Lemma 97. We have for the decryption advantage of* Dec' *from Algorithm 15:*

$$\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') = \left(1 - \frac{1}{Q} \cdot \binom{m+d_2}{d_2} - \varepsilon_{\mathsf{fh}}\right) \cdot \mathsf{adv}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}). \tag{2.466}$$

Proof. Draw $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$, $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q \leftarrow \mathsf{KeyGen}(\mathsf{msk}, 0)$ and $\mathsf{sk}_* \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_{*,\lambda})$. We have

$$\Pr\Big[\mathsf{sk}_* \in \mathrm{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1, \ldots, \mathsf{sk}_Q\}\Big] = 1 - \frac{1}{Q} \cdot \binom{m+d_2}{d_2} - \varepsilon_{\mathsf{fh}}. \tag{2.467}$$

Now, let $\mathsf{msk}' = (\mathsf{msk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_Q, \mathsf{sk}_*)$ and $\mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x)$ be the input of Dec' (for some $x \in \mathcal{X}'_\lambda$). Note that ct' is of the shape

$$\mathsf{ct}' = (\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_Q(\mathsf{ct})) \tag{2.468}$$

for $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, \nu(x))$. We distinguish two cases:

If sk_* lies in the span of $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$, then Dec' computes values $\alpha_1, \ldots, \alpha_Q \in \mathbb{Z}_q$ s.t. $\mathsf{sk}_* = \alpha_1 \mathsf{sk}_1 + \ldots + \alpha_Q \mathsf{sk}_Q$. It uses $\alpha_1, \ldots, \alpha_q$ to compute

$$\mathsf{sk}_*(\mathsf{ct}) = (\alpha_1 \cdot \mathsf{sk}_1 + \ldots + \alpha_Q \cdot \mathsf{sk}_Q)(\mathsf{ct}) \tag{2.469}$$

$$= \alpha_1 \cdot \mathsf{sk}_1(\mathsf{ct}) + \ldots + \alpha_Q \cdot \mathsf{sk}_1(\mathsf{ct}) = \alpha^T \cdot \mathsf{ct}' \tag{2.470}$$

and uses $\mathsf{sk}_*(\mathsf{ct})$ to output $\mathsf{Dec}(\mathsf{sk}_*, \mathsf{ct})$. In this case, the output of $\mathsf{Dec}'(\mathsf{msk}', \mathsf{ct}')$ is equally distributed as $\mathsf{Dec}(\mathsf{sk}_*, \mathsf{ct})$.

If sk_* does not lie in span of $\mathsf{sk}_1, \ldots, \mathsf{sk}_Q$, then Dec' outputs a uniformly random number of $\mathbb{Z}_p$. In this case, the probability of Dec' to output x is $1/p$.

In total, we have for decryption probability of Dec'

$$\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') = \mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}) \cdot \Pr\Big[\mathsf{sk}_* \in \mathrm{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1, \ldots, \mathsf{sk}_Q\}\Big] \tag{2.471}$$

$$+ \frac{1}{p} \cdot \Pr\Big[\mathsf{sk}_* \notin \mathrm{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1, \ldots, \mathsf{sk}_Q\}\Big] \tag{2.472}$$

$$= \mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}) \cdot \left(1 - \frac{1}{Q} \cdot \binom{m+d_2}{d_2} - \varepsilon_{\mathsf{fh}}\right) \tag{2.473}$$

$$+ \frac{1}{p} \cdot \left(\frac{1}{Q} \cdot \binom{m+d_2}{d_2} + \varepsilon_{\mathsf{fh}}\right). \tag{2.474}$$

For the decryption advantage, it follows

$$\begin{aligned}
&\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') && (2.475)\\
=&\frac{\#\mathcal{X}'_\lambda \cdot \mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') - 1}{\#\mathcal{X}'_\lambda - 1} && (2.476)\\
=&\frac{p \cdot \mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') - 1}{p-1} && (2.477)\\
=&\frac{p \cdot \mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}) \cdot \left(1 - \frac{1}{Q} \cdot \binom{m+d_2}{d_2} - \varepsilon_{\mathsf{fh}}\right) - 1}{p-1} + \frac{\frac{1}{Q} \cdot \binom{m+d_2}{d_2} + \varepsilon_{\mathsf{fh}}}{p-1} && (2.478)\\
=&\frac{(p \cdot \mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}) - 1) \cdot \left(1 - \frac{1}{Q} \cdot \binom{m+d_2}{d_2} - \varepsilon_{\mathsf{fh}}\right)}{p-1} && (2.479)\\
&- \frac{\frac{1}{Q} \cdot \binom{m+d_2}{d_2} + \varepsilon_{\mathsf{fh}}}{p-1} + \frac{\frac{1}{Q} \cdot \binom{m+d_2}{d_2} + \varepsilon_{\mathsf{fh}}}{p-1} && (2.480)\\
=&\frac{p \cdot \mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}) - 1}{p-1} \cdot \left(1 - \frac{1}{Q} \cdot \binom{m+d_2}{d_2} - \varepsilon_{\mathsf{fh}}\right) && (2.481)\\
=&\mathsf{adv}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}) \cdot \left(1 - \frac{1}{Q} \cdot \binom{m+d_2}{d_2} - \varepsilon_{\mathsf{fh}}\right). && \square
\end{aligned}$$

Proof Theorem 94. Let $Q \in \mathbb{N}$ and set

$$\begin{aligned}
\varepsilon_{\mathsf{fh}} :=& 1 - \frac{1}{Q+1}\binom{m+d_2}{d_2} && (2.482)\\
&- \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)\\ \mathsf{sk}_1,\ldots,\mathsf{sk}_Q \leftarrow \mathsf{KeyGen}(\mathsf{msk},0)\\ \mathsf{sk}_* \leftarrow \mathsf{KeyGen}(\mathsf{msk},f_{*,\lambda})}} \Big[\mathsf{sk}_* \in \operatorname{span}_{\mathbb{Z}_q}\{\mathsf{sk}_1,\ldots,\mathsf{sk}_Q\}\Big]. && (2.483)
\end{aligned}$$

According to Lemma 97, there is an adversary $\mathcal{A}^{\mathsf{fh}}$ on the function-hiding security of FE that makes Q encryption queries and $\Theta(Q \cdot m^{d_2})$ arithmetic operations. For the advantage of $\mathcal{A}^{\mathsf{fh}}$ in Game 4, we have

$$\mathsf{adv}^{\mathsf{FH}}_{\mathsf{FE}}(\mathcal{A}^{\mathsf{fh}}) \geq \varepsilon_{\mathsf{fh}}. \tag{2.484}$$

This shows the first claim.

For the second claim, let $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \mathsf{Dec}')$ be the SKE from Algorithm 15. Lemma 98 implies that we have

$$\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') = \left(1 - \frac{1}{Q} \cdot \binom{m+d_2}{d_2} - \varepsilon_{\mathsf{fh}}\right) \cdot \mathsf{adv}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}). \tag{2.485}$$

With Theorem 93 the existence of an adversary $\mathcal{A}^{\mathsf{ind\text{-}cpa}}$ against the selective IND-CPA security of FE now follows. Since the time complexity of $\mathsf{Setup}'_{\mathsf{Pre}}$ lies in $O(nQ)$, $\mathcal{A}^{\mathsf{ind\text{-}cpa}}$ makes $3N$ encryption, Q function queries and $O(m^{d_2} \cdot QN)$ arithmetic operations over $\mathbb{Z}_p, \mathbb{Z}_q$ and $\mathbb{Z}$.

For the advantage of $\mathcal{A}^{\textsf{ind-cpa}}$, we have

$$
\begin{aligned}
& \mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}) && (2.486)\\
\overset{\text{Theorem 93}}{\geq}\ & \frac{\mathsf{adv}_{\mathsf{SKE}'}^{\mathsf{dec}}(\mathsf{Dec}') - (d+2)\cdot \varepsilon_{\mathsf{width}}}{4dQ} && (2.487)\\
& - 16\exp(-2N') - (d+4)\cdot N \cdot \varepsilon_{\mathsf{width}} && (2.488)\\
\overset{\text{Lemma 98}}{\geq}\ & \frac{\left(1 - \frac{1}{Q}\cdot \binom{m+d_2}{d_2} - \varepsilon_{\mathsf{fh}}\right)\cdot \mathsf{adv}_{\mathsf{FE}}^{\mathsf{dec}}(\mathsf{Dec}) - (d+2)\cdot \varepsilon_{\mathsf{width}}}{4dQ} && (2.489)\\
& - 16\exp(-2N') - (d+4)\cdot N \cdot \varepsilon_{\mathsf{width}}. && \square
\end{aligned}
$$

2.5.3 Homogeneity Among Ciphertexts

In this subsection, we will prove that ciphertexts of functional encryption schemes must be homogeneous in a certain algebraic sense. For this end, let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be an arbitrary FE scheme. In this subsection, we do not require that FE is lattice-based or supports a specific functionality. We only need the following:

1. Each ciphertext of FE lies in $\mathbb{Z}_q^m$ for a prime q.
2. Each secret key of FE is a polynomial in $\mathbb{Z}_q[C_1, \ldots, C_m]$ of degree d.
3. KeyGen is *deterministic*[2].

Let $d \in \mathbb{N}$ be constant, $m, Q \in \mathsf{poly}(\lambda)$ s.t. $Q \in \omega(m)$ and $m \geq \lambda$. Let $D \in O((m^d/Q)^{1/(d-1)})$ be the function from Theorem 10 for Q polynomials of degree d over m variables. Set

$$t := \binom{Q+D}{D} \in Q^{O(D)}. \tag{2.490}$$

Note that D is sublinear in m, hence t is subexponential in m.

We aim to be as general as possible regarding the function D in this subsection. For this end, we will formulate and prove the Lemmas 100 to 106 with respect to the function classes $\mathsf{negl}(t)$ and $\mathsf{poly}(t)$ (which coincide with $\mathsf{negl}(\lambda)$ and $\mathsf{poly}(\lambda)$ if D is constant). Unfortunately, the current proof strategy for Theorem 99 relies on Lemma 107, which can only been proven for constant D. Because of this, Theorem 99 only claims homogeneity for constant D, however, in Conjecture 2, I conjecture that the claim of Theorem 99 can be extended to non-constant D by improving the proof strategy in this subsection.

Now, for each $\lambda \in \mathbb{N}$, fix a collection $f_{1,\lambda}, \ldots, f_{Q(\lambda),\lambda} \in \mathcal{F}_\lambda$. For ease of notation, we will omit the λ-subscript this time and just write $f_1, \ldots, f_Q$ when we mean the series $((f_{1,\lambda}, \ldots, f_{Q(\lambda),\lambda}))_\lambda$. Since KeyGen is deterministic, we can map each msk deterministically to the collection

$$\mathsf{sk}_1 := \mathsf{KeyGen}(\mathsf{msk}, f_1), \ldots, \mathsf{sk}_Q := \mathsf{KeyGen}(\mathsf{msk}, f_Q) \tag{2.491}$$

of secret keys. Because each secret key is of degree d, Theorem 10 implies the

[2] We will show in Section 2.5.4, how we can turn any randomized key generation algorithm into a deterministic one without loss of generality.

existence of a polynomial $h \in \mathbb{Z}_q[S_1, \dots, S_Q]$ with

$$h \neq 0, \tag{2.492}$$

$$\deg h \leq D, \tag{2.493}$$

$$h(\mathsf{sk}_1, \dots, \mathsf{sk}_Q) = 0 \in \mathbb{Z}_q[C]. \tag{2.494}$$

Since the secret keys $\mathsf{sk}_1, \dots, \mathsf{sk}_Q$ depend deterministically on msk, we can choose for each msk in a deterministic way a polynomial h_{msk} that fulfils Eqs. (2.492) to (2.494).

Finally, let $\widetilde{\mathcal{X}} \subset \mathcal{X}$ be a subspace of messages s.t. the size of $\widetilde{\mathcal{X}}$ is polynomial in t, i.e. $\#\widetilde{\mathcal{X}}_\lambda \in \mathsf{poly}(t)$, and there is a deterministic algorithm that on input 1^λ can enumerate the set $\widetilde{\mathcal{X}}_\lambda$ by making $\mathsf{poly}(t)$ arithmetic operations over $\mathbb{Z}_p$ for some $p < q$.

Before we can state the main theorem of this subsection, we need to introduce the following notion:

Definition 36. For a fixed master secret key msk and a subset $A \subset [Q(\lambda)]$, denote by $\tau_{\mathsf{msk},A} \colon \mathbb{Z}_q[S] \to \mathbb{Z}_q[S, C]$ the ring morphism that substitutes S_i by $\mathsf{sk}_i(C)$ iff $i \in A$, i.e.,

$$\begin{aligned} \tau_{\mathsf{msk},A} \colon \mathbb{Z}_q[S_1, \dots, S_Q] &\longrightarrow \mathbb{Z}_q[S_1, \dots, S_Q, C_1, \dots, C_m] \\ S_i &\longmapsto \begin{cases} \mathsf{sk}_i(C), & i \in A, \\ S_i, & i \notin A. \end{cases} \end{aligned}$$

Theorem 99 (Homogeneity among Ciphertexts)**.** *Let* FE *be selectively IND-CPA secure and assume that*

$$d > 1, \qquad m \geq \lambda, \qquad \text{and} \qquad Q \in \Omega(m^d). \tag{2.495}$$

Note that in this case $D \in O\Big(\big(m^d/Q\big)^{1/(d-1)}\Big) = O(1)$ *from Theorem 10 is bounded by a constant.*

Then, there is a set G*, maps* $A, B : \mathbb{N} \to \mathcal{P}(\mathbb{N})$ *and functions* $\varepsilon \in \mathsf{negl}(\lambda)$*,* $\rho \notin \mathsf{negl}(\lambda)$ *s.t. we have for all* $\lambda \in \mathbb{N}$*:*

1. $A(\lambda) \subset B(\lambda) \subseteq [Q(\lambda)]$ *and* $\#B(\lambda) = \#A(\lambda) + 1$*.*
2. $\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)}[\mathsf{msk} \in G] \geq t(\lambda)^{-1} \geq Q(\lambda)^{-D}$*.*
3. *For all* $x \in \widetilde{\mathcal{X}}_\lambda$*, we have*

$$\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)}} \Big[\tau_{\mathsf{msk},B(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0 \mid \mathsf{msk} \in G\Big] \geq 1 - \varepsilon(\lambda), \tag{2.496}$$

$$\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)}} \Big[\tau_{\mathsf{msk},A(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0 \mid \mathsf{msk} \in G\Big] \leq 1 - \rho(\lambda). \tag{2.497}$$

As explained above, I conjecture that the claim of Theorem 99 can be extended to hold even for non-constant $D \in \omega(1)$ if FE is secure against $\mathsf{poly}(t)$-adversaries.

Conjecture 2 (Subexponential Homogeneity among Ciphertexts)**.** *Assume that each adversary against the selective IND-CPA security of* FE *that makes* $\mathsf{poly}(t)$

encryption queries and arithmetic operations over $\mathbb{Z}_p$ *and* $\mathbb{Z}_p$ *has an advantage of at most* $\mathsf{negl}(t)$.

Then, there are a set G, *maps* $A, B : \mathbb{N} \to \mathcal{P}(\mathbb{N})$ *and functions* $\varepsilon \in \mathsf{negl}(t)$, $\rho \notin \mathsf{negl}(t)$ *s.t. we have for all* $\lambda \in \mathbb{N}$:

1. $A(\lambda) \subset B(\lambda) \subseteq [Q(\lambda)]$ *and* $\#B(\lambda) = \#A(\lambda) + 1$.
2. $\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)}[\mathsf{msk} \in G] \geq t(\lambda)^{-1} \geq Q(\lambda)^{-D(\lambda)}$.
3. *For all* $x \in \widetilde{\mathcal{X}}_\lambda$, *we have*

$$\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)}} \left[\tau_{\mathsf{msk}, B(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0 \mid \mathsf{msk} \in G\right] \geq 1 - \varepsilon(\lambda), \tag{2.498}$$

$$\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)}} \left[\tau_{\mathsf{msk}, A(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0 \mid \mathsf{msk} \in G\right] \leq 1 - \rho(\lambda). \tag{2.499}$$

Theorem 99 essentially states that for each $\lambda \in \mathbb{N}$ there is an index $i_\dagger$, $\{i_\dagger\} = B(\lambda) \setminus A(\lambda)$, s.t. for $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$ there exists a polynomial $h \in \mathbb{Z}_q[S_i \mid i \in B(\lambda)]$ s.t. $h((\mathsf{sk}_i(\mathsf{ct}))_{i \in B(\lambda)})$ vanishes on almost all ciphertexts $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)$ for all $x \in \widetilde{\mathcal{X}}_\lambda$, while $h((\mathsf{sk}_i(\mathsf{ct}))_{i \in A(\lambda)}, S_{i_\dagger})$ will not vanish with noticeable probability. This means, in a non-negligible number of cases the value of $\mathsf{sk}_{i_\dagger}(\mathsf{ct})$ is a root of the non-zero polynomial $h((\mathsf{sk}_i(\mathsf{ct}))_{i \in A(\lambda)}, S_{i_\dagger})$ of degree $\leq D$ whose coefficients are polynomials in $(\mathsf{sk}_i(\mathsf{ct}))_{i \in A(\lambda)}$ and $(S_i)_{i \notin B(\lambda)}$ and which is univariate in the variable $S_{i_\dagger}$. In particular, given $(\mathsf{sk}_i(\mathsf{ct}))_{i \in A(\lambda)}$, there are at most D different values that the evaluation $\mathsf{sk}_{i_\dagger}(\mathsf{ct})$ can take. Our computationally unbounded decryptor (resp. extractor) Dec' in Section 2.5.4 will use this fact to restrict the number of possible decryptions to D from $p \in \omega(D)$. This will give it a noticeable advantage at decryption, which leads to an efficient adversary on a compact FE scheme FE.

Let us assume in the following that each adversary against the selective IND-CPA security of FE of time complexity $\mathsf{poly}(t)$ has an advantage in $\mathsf{negl}(t)$.

Lemma 100. *There exists a map* $\mathcal{I}\colon \mathbb{N} \to P(\mathbb{N})$ *s.t.*

$$\forall \lambda \in \mathbb{N} : \ \mathcal{I}(\lambda) \subseteq [Q(\lambda)] \quad \textit{and} \quad \#\mathcal{I}(\lambda) = D(\lambda).$$

Additionally, the probability when we sample $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$ *that* h_{msk} *contains non-trivially a monomial* $S_{i_1} \cdots S_{i_{D'}}$ *for some* $D' \leq D$ *with* $i_1, \ldots, i_{D'} \in \mathcal{I}(\lambda)$ *is larger than* t^{-1}.

Proof. For each msk, h_{msk} must be a non-zero polynomial in $\mathbb{Z}_q[S_1, \ldots, S_Q]$ of degree $\leq D$. Since $\mathbb{Z}_q[S_1, \ldots, S_Q]$ contains $t = \binom{Q+D}{D}$ monomials of degree $\leq D$, there must exist one monomial $S_{i_1} \cdots S_{i_{D'}}$ for each $\lambda \in \mathbb{N}$ s.t.

$$\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)} \left[h \text{ contains } S_{i_1} \cdots S_{i_{D'}}\right] \geq \frac{1}{t}.$$

Hence, we can choose $\mathcal{I}(\lambda)$ s.t. it contains $i_1, \ldots, i_{D'}$. □

We will call a master secret key msk **good**, if h_{msk} contains non-trivially a monomial $S_{i_1} \cdots S_{i_{D'}}$ with $i_1, \ldots, i_{D'} \in \mathcal{I}(\lambda)$, and we will call msk **bad**, otherwise. If we set

$$G := \{\mathsf{msk} \mid \mathsf{msk} \text{ is a good master secret key for } \mathsf{FE}\}, \tag{2.500}$$

then we have

$$\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)}[\mathsf{msk} \in G] \geq t^{-1}. \tag{2.501}$$

Denote by $\mathsf{Setup}_{\mathsf{good}}(1^\lambda)$ the distribution $\mathsf{Setup}(1^\lambda)$ conditioned on the set G.

Lemma 101. *Let $d, D, m, Q \in \mathbb{N}$ be arbitrary. Set $L := \binom{m+dD}{dD}$ and let $Y^{\alpha_1}, \ldots, Y^{\alpha_L}$ be an enumeration of all monomials of $\mathbb{Z}_q[C_1, \ldots, C_m]$ of degree $\leq dD$. Let*

$$\begin{aligned} \psi_{dD} : \mathbb{Z}_q^m &\longrightarrow \mathbb{Z}_q^L & (2.502) \\ c &\longmapsto (c^{\alpha_1}, \ldots, c^{\alpha_L}) & (2.503) \end{aligned}$$

be the map that assigns to each point c a vector of all products of its entries of degree $\leq dD$.

We have for all $\ell \in \mathbb{N}, c_1, \ldots, c_{\ell+1} \in \mathbb{Z}_q^m$ and $h \in \mathbb{Z}_q[C_1, \ldots, C_m, S_1, \ldots, S_Q]$ of degree $\leq dD$ the following implication,

$$\left.\begin{array}{l} \psi_{dD}(c_{\ell+1}) \in \operatorname{span}_{\mathbb{Z}_q}\{\psi_{dD}(c_1), \ldots, \psi_{dD}(c_\ell)\} \\ \textit{and } \ \forall i \in [\ell] \colon h(c_i, S_1, \ldots, S_Q) = 0 \end{array}\right\} \implies h(c_{\ell+1}, S_1, \ldots, S_Q) = 0.$$

Proof. Since $h \in \mathbb{Z}_q[C, S]$ is of degree $\leq D$, there are polynomials $h_1, \ldots, h_L \in \mathbb{Z}_q[S]$ s.t. h can be written as

$$h(C, S) = \sum_{i=1}^{L} h_i(S) \cdot C^{\alpha_i}. \tag{2.504}$$

Assume that we have $\psi_{dD}(c_{\ell+1}) \in \operatorname{span}_{\mathbb{Z}_q}\{\psi_{dD}(c_1), \ldots, \psi_{dD}(c_\ell)\}$ and $h(c_i, S) = 0$ for each $i \in [\ell]$. Then, there are scalars $\gamma_1, \ldots, \gamma_\ell \in \mathbb{Z}_q$ s.t.

$$\psi_{dD}(c_{\ell+1}) = \gamma_1 \cdot \psi_{dD}(c_1) + \ldots + \gamma_\ell \cdot \psi_{dD}(c_\ell). \tag{2.505}$$

In particular, we have for each multi-index α_i

$$c_{\ell+1}^{\alpha_i} = \gamma_1 \cdot c_1^{\alpha_i} + \ldots + \gamma_\ell \cdot c_\ell^{\alpha_i}. \tag{2.506}$$

We now have

$$\begin{aligned} h(c_{\ell+1}, S) &= \sum_{i=1}^{L} h_i(S) \cdot c_{\ell+1}^{\alpha_i} = \sum_{i=1}^{L} h_i(S) \cdot \left(\sum_{j=1}^{\ell} \gamma_j c_j^{\alpha_i} \right) & (2.507) \\ &= \sum_{j=1}^{\ell} \gamma_j \cdot \left(\sum_{i=1}^{L} h_i(S) \cdot c_j^{\alpha_i} \right) = \sum_{j=1}^{\ell} \gamma_j \cdot h(c_j, S) & (2.508) \\ &= \sum_{j=1}^{\ell} \gamma_j \cdot 0 = 0. \end{aligned}$$

□

We will use the map ϕ_{dD} from Lemma 101 to construct an adversary that can detect algebraic differences between ciphertexts.

Algorithm 16. We will give here an adversary $\mathcal{A}$ on the selective IND-CPA security of FE. $\mathcal{A}$ is parametrized by a parameter $\ell \in \mathsf{poly}(t)$:

Step 1: $\mathcal{A}$ samples $y, z \leftarrow \widetilde{\mathcal{X}}_\lambda$.

Step 2: $\mathcal{A}$ defines two lists $(x_i^{(0)})_{i=1,\dots,\ell+1}$ and $(x_i^{(1)})_{i=1,\dots,\ell+1}$ by

$$x_i^{(0)} := y \text{ and } x_i^{(1)} := \begin{cases} y, & \text{if } i \in [\ell], \\ z, & \text{if } i = \ell+1. \end{cases}$$

Step 3: $\mathcal{A}$ submits both lists to the challenger and receives a list of ciphertexts $\mathsf{ct}_1, \dots, \mathsf{ct}_\ell$ of y and $\mathsf{ct}_{\ell+1}$ of $x_{\ell+1}^{(b)}$ for unknown $b \in \{0,1\}$.

Step 4: Let $\psi_{dD} \colon \mathbb{Z}_q^m \to \mathbb{Z}_q^L$ be the map from Lemma 101. $\mathcal{A}$ computes

$$V := \operatorname{span}_{\mathbb{Z}_q}\{\psi_{dD}(\mathsf{ct}_1), \dots, \psi_{dD}(\mathsf{ct}_\ell)\} \subseteq \mathbb{Z}_q^L.$$

Step 5: If $\psi_{dD}(\mathsf{ct}_{\ell+1}) \in V$, then $\mathcal{A}$ outputs $b = 0$. Otherwise, $\mathcal{A}$ outputs $b = 1$.

Lemma 102. *Adversary $\mathcal{A}$ from Algorithm 16 makes $\ell+1$ ciphertext queries and $O(\ell^2 \cdot \binom{m+dD}{dD}) + \mathsf{poly}(t)$ arithmetic operations over $\mathbb{Z}_p$ and $\mathbb{Z}_q$.*

Proof. $\mathcal{A}$'s runtime is dominated by enumerating $\widetilde{\mathcal{X}}_\lambda$ in Step 1 and by checking $\psi_{dD}(\mathsf{ct}_{\ell+1}) \in V$ in Step 5. Enumerating $\widetilde{\mathcal{X}}_\lambda$ costs $\mathsf{poly}(t)$ arithmetic operations over $\mathbb{Z}_p$. Checking if $\psi_{dD}(\mathsf{ct}_{\ell+1})$ is linearly dependent from $\psi_{dD}(\mathsf{ct}_1), \dots, \psi_{dD}(\mathsf{ct}_\ell)$ makes it necessary to perform Gaussian elimination on a matrix of shape $\ell \times \binom{m+dD}{dD}$, which costs $O(\ell^2 \cdot \binom{m+dD}{dD})$ operations over $\mathbb{Z}_q$. □

Lemma 103. *Denote by $\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}|\mathsf{msk}, y, z)$ the advantage of $\mathcal{A}$ for some fixed λ conditioned on the event that the challenger draws the master secret key* msk *and $\mathcal{A}$ draws the messages y, z in Step 1.*

We have for all msk*, $y, z \in \widetilde{\mathcal{X}}_\lambda$*

$$\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}|\mathsf{msk}, y, z) \geq -\frac{1}{\ell+1} \cdot \binom{m+dD}{dD}. \tag{2.509}$$

Proof. We will only lower bound the probability that $\mathcal{A}$ outputs 0 whenever the secret bit b of the challenger is 0. In this case, the received ciphertexts $\mathsf{ct}_1, \dots, \mathsf{ct}_{\ell+1}$ are all from the same distribution $\mathsf{Enc}(\mathsf{msk}, y)$. Because of Lemma 96 about learning vector spaces, we have

$$\Pr_{\mathsf{ct}_1,\dots,\mathsf{ct}_{\ell+1} \leftarrow \mathsf{Enc}(\mathsf{msk},y)}[\psi_{dD}(\mathsf{ct}_{\ell+1}) \in \operatorname{span}_{\mathbb{Z}_q}\{\psi_{dD}(\mathsf{ct}_1), \dots, \psi_{dD}(\mathsf{ct}_\ell)\}] \tag{2.510}$$

$$\geq 1 - \frac{1}{\ell+1} \cdot \binom{m+dD}{dD}, \tag{2.511}$$

since each $\psi_{dD}(\mathsf{ct}_i)$ lies in $\mathbb{Z}_q^{\binom{m+dD}{dD}}$. Hence, the advantage of $\mathcal{A}$ must be at least $-\binom{m+dD}{dD}/t$. □

Lemma 104. *For each $\ell \in \mathsf{poly}(t)$, there exists an $\varepsilon_\ell \in \mathsf{negl}(t)$ s.t. we have for each pair of sequences $(y_\lambda)_\lambda, (z_\lambda)_\lambda \in \widetilde{\mathcal{X}}$*

$$\begin{aligned}&\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda)\\ \mathsf{ct}_1,\ldots,\mathsf{ct}_{\ell'(\lambda)} \leftarrow \mathsf{Enc}(\mathsf{msk},y_\lambda)\\ \mathsf{ct}_z \leftarrow \mathsf{Enc}(\mathsf{msk},z_\lambda)}} \Big[\psi_{dD}(\mathsf{ct}_z) \in \operatorname{span}_{\mathbb{Z}_q}\big\{\psi_{dD}(\mathsf{ct}_1), \ldots, \psi_{dD}(\mathsf{ct}_{\ell'(\lambda)})\big\}\Big]\\ &\geq 1 - \frac{1}{\ell} - \varepsilon_\ell(\lambda)\end{aligned}$$

for

$$\ell' = \ell \cdot \binom{m+dD}{dD} \cdot t \cdot \#\widetilde{\mathcal{X}}_\lambda^2 - 1 \in \mathsf{poly}(t). \tag{2.512}$$

Proof. In the IND-CPA Game 3 between a challenger $\mathcal{C}$ and adversary $\mathcal{A}$ from Algorithm 16, denote by event the event that the master secret key msk sampled by $\mathcal{C}$ is good and $y_\lambda, z_\lambda \in \widetilde{\mathcal{X}}_\lambda$ have been drawn by $\mathcal{A}$ in Step 1. Denote the advantage of $\mathcal{A}$ conditioned on event and $\neg\mathsf{event}$ by $\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}|\mathsf{event})$ and $\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}|\neg\mathsf{event})$, respectively.

Instantiate $\mathcal{A}$ with parameter ℓ'. We have for the advantage of $\mathcal{A}$

$$\begin{aligned}&\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A})\\ =&\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}|\mathsf{event}) \cdot \Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda), y', z' \leftarrow \widetilde{\mathcal{X}}_\lambda}[\mathsf{msk} \in G, y' = y_\lambda, z' = z_\lambda]\\ &+ \mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}|\neg\mathsf{event}) \cdot \Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)}[\mathsf{msk} \notin G \wedge y' \neq y_\lambda, z' \neq z_\lambda]\\ \overset{\text{Lemma 103}}{\geq}&\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}|\mathsf{event}) \cdot \Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda), y', z' \leftarrow \widetilde{\mathcal{X}}_\lambda}[\mathsf{msk} \in G, y' = y_\lambda, z' = z_\lambda]\\ &+ \left(-\frac{\binom{m+dD}{dD}}{\ell'+1}\right) \cdot \Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)}[\mathsf{msk} \notin G \wedge y' \neq y_\lambda, z' \neq z_\lambda]\\ \overset{\text{Eq. (2.501)}}{\geq}&\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}|\mathsf{event}) \cdot \frac{1}{t \cdot \#\widetilde{\mathcal{X}}_\lambda^2} + \left(-\frac{\binom{m+dD}{dD}}{\ell'+1}\right) \cdot \frac{t \cdot \#\widetilde{\mathcal{X}}_\lambda^2 - 1}{t \cdot \#\widetilde{\mathcal{X}}_\lambda^2}.\end{aligned}$$

We demanded at the beginning at this subsection that the advantage of each adversary with time complexity $\mathsf{poly}(t)$ against the selective IND-CPA security of FE must lie in $\mathsf{negl}(t)$. Hence, there is an $\varepsilon'_\ell \in \mathsf{negl}(t)$ s.t.

$$\varepsilon'_\ell \geq \mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}|\mathsf{event}) \cdot \frac{1}{t \cdot \#\widetilde{\mathcal{X}}_\lambda^2} - \frac{\binom{m+dD}{dD}}{\ell'+1} \cdot \frac{t \cdot \#\widetilde{\mathcal{X}}_\lambda^2 - 1}{t \cdot \#\widetilde{\mathcal{X}}_\lambda^2}. \tag{2.513}$$

We rewrite this inequality as

$$\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}|\mathsf{event}) \leq t \cdot \#\widetilde{\mathcal{X}}_\lambda^2 \cdot \varepsilon'_\ell + \frac{(t \cdot \#\widetilde{\mathcal{X}}_\lambda^2 - 1) \cdot \binom{m+dD}{dD}}{\ell'+1}. \tag{2.514}$$

Let us now inspect the advantage of $\mathcal{A}$ conditioned on event. $\mathcal{A}$ outputs 0 if $\psi_{dD}(\mathsf{ct}_{\ell'+1}) \in V := \operatorname{span}_{\mathbb{Z}_q}\{\psi_{dD}(\mathsf{ct}_1), \ldots, \psi_{dD}(\mathsf{ct}_{\ell'})\}$ and 1 if $\psi_{dD}(\mathsf{ct}_{\ell'+1}) \notin V$ for $\mathsf{ct}_1, \ldots, \mathsf{ct}_{\ell'+1} \leftarrow \mathsf{Enc}(\mathsf{msk}, y_\lambda)$ and $\mathsf{ct}_{\ell'+1} \leftarrow \mathsf{Enc}(\mathsf{msk}, x_b)$ for $b \leftarrow \{0,1\}$

where $x_0 = y_\lambda$ and $x_1 = z_\lambda$. We have

$$\begin{aligned}
&\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}|\mathsf{event})\\
&= \Pr_{\substack{\mathsf{msk}\leftarrow\mathsf{Setup}_{\mathsf{good}}(1^\lambda)\\ \mathsf{ct}_1,\ldots,\mathsf{ct}_{\ell'}\leftarrow\mathsf{Enc}(\mathsf{msk},y_\lambda)\\ \mathsf{ct}_y\leftarrow\mathsf{Enc}(\mathsf{msk},y_\lambda)}}[\psi_{dD}(\mathsf{ct}_y)\in V] + \Pr_{\substack{\mathsf{msk}\leftarrow\mathsf{Setup}_{\mathsf{good}}(1^\lambda)\\ \mathsf{ct}_1,\ldots,\mathsf{ct}_{\ell'}\leftarrow\mathsf{Enc}(\mathsf{msk},y_\lambda)\\ \mathsf{ct}_y\leftarrow\mathsf{Enc}(\mathsf{msk},z_\lambda)}}[\psi_{dD}(\mathsf{ct}_z)\notin V] - 1\\
&= \Pr_{\substack{\mathsf{msk}\leftarrow\mathsf{Setup}_{\mathsf{good}}(1^\lambda)\\ \mathsf{ct}_1,\ldots,\mathsf{ct}_{\ell'}\leftarrow\mathsf{Enc}(\mathsf{msk},y_\lambda)\\ \mathsf{ct}_y\leftarrow\mathsf{Enc}(\mathsf{msk},y_\lambda)}}[\psi_{dD}(\mathsf{ct}_y)\in V] - \Pr_{\substack{\mathsf{msk}\leftarrow\mathsf{Setup}_{\mathsf{good}}(1^\lambda)\\ \mathsf{ct}_1,\ldots,\mathsf{ct}_{\ell'}\leftarrow\mathsf{Enc}(\mathsf{msk},y_\lambda)\\ \mathsf{ct}_y\leftarrow\mathsf{Enc}(\mathsf{msk},z_\lambda)}}[\psi_{dD}(\mathsf{ct}_z)\in V]
\end{aligned}$$

Set

$$\alpha := \Pr_{\substack{\mathsf{msk}\leftarrow\mathsf{Setup}_{\mathsf{good}}(1^\lambda)\\ \mathsf{ct}_1,\ldots,\mathsf{ct}_{\ell'}\leftarrow\mathsf{Enc}(\mathsf{msk},y_\lambda)\\ \mathsf{ct}_y\leftarrow\mathsf{Enc}(\mathsf{msk},z_\lambda)}}[\psi_{dD}(\mathsf{ct}_z)\in V] \tag{2.515}$$

and note that we have

$$\Pr_{\substack{\mathsf{msk}\leftarrow\mathsf{Setup}_{\mathsf{good}}(1^\lambda)\\ \mathsf{ct}_1,\ldots,\mathsf{ct}_{\ell'}\leftarrow\mathsf{Enc}(\mathsf{msk},y_\lambda)\\ \mathsf{ct}_y\leftarrow\mathsf{Enc}(\mathsf{msk},y_\lambda)}}[\psi_{dD}(\mathsf{ct}_y)\in V] \geq 1 - \frac{1}{\ell'+1}\cdot\binom{m+dD}{dD} \tag{2.516}$$

because of Lemma 96. We then have for $\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}|\mathsf{event})$

$$\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}|\mathsf{event}) \geq 1 - \frac{\binom{m+dD}{dD}}{\ell'+1} - \alpha. \tag{2.517}$$

When we combine this with Eq. (2.514), we get

$$t\cdot\#\widetilde{\mathcal{X}}_\lambda^2\cdot\varepsilon'_\ell + \frac{(t\cdot\#\widetilde{\mathcal{X}}_\lambda^2-1)\cdot\binom{m+dD}{dD}}{\ell'+1} \geq 1 - \frac{\binom{m+dD}{dD}}{\ell'+1} - \alpha, \tag{2.518}$$

which is equivalent to

$$\alpha \geq 1 - t\cdot\#\widetilde{\mathcal{X}}_\lambda^2\cdot\frac{\binom{m+dD}{dD}}{\ell'+1} + t\cdot\#\widetilde{\mathcal{X}}_\lambda^2\cdot\varepsilon'_\ell. \tag{2.519}$$

The claim now follows by setting

$$\varepsilon_\ell := t\cdot\#\widetilde{\mathcal{X}}_\lambda^2\cdot\varepsilon'_\ell \in \mathsf{negl}(t) \tag{2.520}$$

and because of

$$\ell' = \ell\cdot\binom{m+dD}{dD}\cdot t\cdot\#\widetilde{\mathcal{X}}_\lambda^2 - 1. \qquad\square$$

Definition 37. For $\lambda\in\mathbb{N}$, $A\subseteq[Q(\lambda)]$ and $x\in\widetilde{\mathcal{X}}_\lambda$, set

$$p_\lambda(A,x) := \Pr_{\substack{\mathsf{msk}\leftarrow\mathsf{Setup}_{\mathsf{good}}(1^\lambda)\\ \mathsf{ct}\leftarrow\mathsf{Enc}(\mathsf{msk},x)}}[\tau_{\mathsf{msk},A}(h_{\mathsf{msk}})(S,\mathsf{ct})=0], \tag{2.521}$$

where $\tau_{\mathsf{msk},A}(h_{\mathsf{msk}})(S,\mathsf{ct})$ results from $h_{\mathsf{msk}}(S_1,\ldots,S_Q)$ by substituting S_i with $\mathsf{sk}_i(\mathsf{ct})$ if $i\in A$.

Lemma 105. *For each $\ell \in \mathsf{poly}(t)$, there exist functions $\ell' \in \mathsf{poly}(t)$, $\ell' \geq \ell$, and $\varepsilon_\ell \in \mathsf{negl}(t)$ s.t. we have for each map $A : \mathbb{N} \to \mathcal{P}(\mathbb{N})$ with $A(\lambda) \subseteq [Q(\lambda)]$ and each pair of sequences $y = (y_\lambda)_\lambda, z = (z_\lambda)_\lambda \in \widetilde{\mathcal{X}}$*

$$p_\lambda(A(\lambda), z_\lambda) \geq \ell'(\lambda) \cdot p_\lambda(A(\lambda), y_\lambda) - (\ell'(\lambda) - 1) - \frac{1}{\ell(\lambda)} - \varepsilon_\ell(\lambda). \tag{2.522}$$

Proof. Set for fixed λ and msk

$$g_{\mathsf{msk}}(C_1, \ldots, C_m) := \tau_{\mathsf{msk},A(\lambda)}(h_{\mathsf{msk}}) \in \mathbb{Z}_q[S][C]. \tag{2.523}$$

We consider g_{msk} as a polynomial with coefficients in $\mathbb{Z}_q[S_1, \ldots, S_Q]$ and variables $C_1, \ldots, C_m$. The degree of g_{msk} is at most $\max_{j \in A(\lambda)}(\deg \mathsf{sk}_j) \cdot \deg h_{\mathsf{msk}} \leq d \cdot D$. Note that we have for $x \in \widetilde{\mathcal{X}}_\lambda$

$$p_\lambda(A(\lambda), x) = \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)}} [g_{\mathsf{msk}}(\mathsf{ct}) = 0]. \tag{2.524}$$

Let $\ell' \in \mathsf{poly}(\lambda)$, $\ell' \geq \ell$ and $\varepsilon_\ell \in \mathsf{negl}(\lambda)$ be the functions from Lemma 104.

For $\mathsf{ct}_z \leftarrow \mathsf{Enc}(\mathsf{msk}, z_\lambda)$, and $\mathsf{ct}_1, \ldots, \mathsf{ct}_{\ell'} \leftarrow \mathsf{Enc}(\mathsf{msk}, y_\lambda)$ we have according to Lemma 101 the following implication of events,

$$\begin{aligned} &\psi_{dD}(\mathsf{ct}_z) \in \operatorname{span}_{\mathbb{Z}_q}\{\psi_{dD}(\mathsf{ct}_1), \ldots, \psi_{dD}(\mathsf{ct}_{\ell'})\}, && (2.525)\\ &g_{\mathsf{msk}}(\mathsf{ct}_1) = \ldots = g_{\mathsf{msk}}(\mathsf{ct}_{\ell'}) = 0 && (2.526)\\ \Longrightarrow\ &g_{\mathsf{msk}}(\mathsf{ct}_z) = 0. && (2.527) \end{aligned}$$

For a fixed msk, we thereby have the following inequalities:

$$\begin{aligned} &\Pr_{\mathsf{ct}_z \leftarrow \mathsf{Enc}(\mathsf{msk}, z_\lambda)} [g_{\mathsf{msk}}(\mathsf{ct}_z) = 0] && (2.528)\\ \geq &\Pr_{\substack{\mathsf{ct}_z \leftarrow \mathsf{Enc}(\mathsf{msk}, z_\lambda) \\ \mathsf{ct}_1, \ldots, \mathsf{ct}_{\ell'} \leftarrow \mathsf{Enc}(\mathsf{msk}, y_\lambda)}} \begin{bmatrix} \psi_{dD}(\mathsf{ct}_z) \in \operatorname{span}_{\mathbb{Z}_q}\{\psi_{dD}(\mathsf{ct}_1), \ldots, \psi_{dD}(\mathsf{ct}_{\ell'})\} \\ g_{\mathsf{msk}}(\mathsf{ct}_1) = \ldots = g_{\mathsf{msk}}(\mathsf{ct}_{\ell'}) = 0 \end{bmatrix} && (2.529)\\ \geq &\Pr_{\substack{\mathsf{ct}_z \leftarrow \mathsf{Enc}(\mathsf{msk}, z_\lambda) \\ \mathsf{ct}_1, \ldots, \mathsf{ct}_{\ell'} \leftarrow \mathsf{Enc}(\mathsf{msk}, y_\lambda)}} \Big[\psi_{dD}(\mathsf{ct}_z) \in \operatorname{span}_{\mathbb{Z}_q}\{\psi_{dD}(\mathsf{ct}_1), \ldots, \psi_{dD}(\mathsf{ct}_{\ell'})\}\Big] && (2.530)\\ &+ \Pr_{\mathsf{ct}_1, \ldots, \mathsf{ct}_{\ell'} \leftarrow \mathsf{Enc}(\mathsf{msk}, y_\lambda)} [g_{\mathsf{msk}}(\mathsf{ct}_1) = \ldots = h_i(\mathsf{ct}_{\ell'}) = 0] - 1 && (2.531)\\ \geq &\Pr_{\substack{\mathsf{ct}_z \leftarrow \mathsf{Enc}(\mathsf{msk}, z_\lambda) \\ \mathsf{ct}_1, \ldots, \mathsf{ct}_{\ell'} \leftarrow \mathsf{Enc}(\mathsf{msk}, y_\lambda)}} \Big[\psi_{dD}(\mathsf{ct}_z) \in \operatorname{span}_{\mathbb{Z}_q}\{\psi_{dD}(\mathsf{ct}_1), \ldots, \psi_{dD}(\mathsf{ct}_{\ell'})\}\Big] && (2.532)\\ &+ \ell' \cdot \Pr_{\mathsf{ct}_y \leftarrow \mathsf{Enc}(\mathsf{msk}, y_\lambda)} [g_{\mathsf{msk}}(\mathsf{ct}_y) = 0] - \ell'. && (2.533) \end{aligned}$$

Note that we used in Eqs. (2.530) and (2.533) the reverse union bound, i.e., $\Pr[A \wedge B] \geq \Pr[A] + \Pr[B] - 1$. We now sample msk according to $\mathsf{Setup}_{\mathsf{good}}(1^\lambda)$, and get

$$p_\lambda(A(\lambda), z_\lambda) = \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct}_z \leftarrow \mathsf{Enc}(\mathsf{msk}, z_\lambda)}} [g_{\mathsf{msk}}(\mathsf{ct}_z) = 0] \tag{2.534}$$

$$\overset{Eq.\ (2.532)}{\geq} \ell' \cdot \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct}_y \leftarrow \mathsf{Enc}(\mathsf{msk}, y_\lambda)}} [g_{\mathsf{msk}}(\mathsf{ct}_y) = 0] - \ell' \tag{2.535}$$

$$+ \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct}_z \leftarrow \mathsf{Enc}(\mathsf{msk}, z_\lambda) \\ \mathsf{ct}_1, \ldots, \mathsf{ct}_{\ell'} \leftarrow \mathsf{Enc}(\mathsf{msk}, y_\lambda)}} \left[\psi_{dD}(\mathsf{ct}_z) \in \operatorname{span}_{\mathbb{Z}_q}\{\psi_{dD}(\mathsf{ct}_1), \ldots, \psi_{dD}(\mathsf{ct}_{\ell'})\}\right] \tag{2.536}$$

$$\overset{Lemma\ 104}{\geq} \ell' \cdot \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct}_y \leftarrow \mathsf{Enc}(\mathsf{msk}, y_\lambda)}} [g_{\mathsf{msk}}(\mathsf{ct}_y) = 0] - \ell' + \left(1 - \frac{1}{\ell} - \varepsilon_\ell\right) \tag{2.537}$$

$$\geq \ell' \cdot \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct}_y \leftarrow \mathsf{Enc}(\mathsf{msk}, y_\lambda)}} [g_{\mathsf{msk}}(\mathsf{ct}_y) = 0] - (\ell' - 1) - \frac{1}{\ell} - \varepsilon_\ell \tag{2.538}$$

$$\geq \ell' \cdot p_\lambda(A(\lambda), y_\lambda) - (\ell' - 1) - \frac{1}{\ell} - \varepsilon_\ell.$$

□

Lemma 106. *Let $A : \mathbb{N} \to \mathcal{P}(\mathbb{N})$ with $A(\lambda) \subseteq [Q(\lambda)]$, and let $(y_\lambda)_\lambda \in \widetilde{\mathcal{X}}$ be a sequence of messages s.t.*

$$p_\lambda(A(\lambda), y_\lambda) \geq 1 - \mathsf{negl}(t). \tag{2.539}$$

Then, there exists an $\varepsilon \in \mathsf{negl}(t)$, s.t. we have for each $\lambda \in \mathbb{N}$ and $x \in \widetilde{\mathcal{X}}_\lambda$

$$p_\lambda(A(\lambda), x) \geq 1 - \varepsilon(t). \tag{2.540}$$

Proof. Let $A : \mathbb{N} \to \mathcal{P}(\mathbb{N})$, $A(\lambda) \subseteq [Q(\lambda)]$, and $(y_\lambda)_\lambda \in \widetilde{\mathcal{X}}$ s.t.

$$\varepsilon'(\lambda) := 1 - p_\lambda(A(\lambda), y_\lambda) \in \mathsf{negl}(t). \tag{2.541}$$

Assume—for the sake of contradiction—that the claim would be incorrect. In this case, there would exist a sequence $(x_\lambda)_\lambda \in \widetilde{\mathcal{X}}$ and a polynomial $r \in \mathsf{poly}(t)$, $r > 0$, s.t. we have for infinitely many $\lambda \in \mathbb{N}$

$$1 - p_\lambda(A(\lambda), x_\lambda) > \frac{1}{r(\lambda)}. \tag{2.542}$$

Lemma 105 now postulates for $\ell = 2 \cdot r$ the existence of an $\ell' \in \mathsf{poly}(t)$ and $\varepsilon_\ell \in \mathsf{negl}(t)$ s.t. we have for all $\lambda \in \mathbb{N}$

$$p_\lambda(A(\lambda), x_\lambda) \geq \ell'(\lambda) \cdot p_\lambda(A(\lambda), y_\lambda) - (\ell'(\lambda) - 1) - \frac{1}{\ell(\lambda)} - \varepsilon_\ell(\lambda) \tag{2.543}$$

$$= \ell'(\lambda) \cdot (1 - \varepsilon'(\lambda)) - (\ell'(\lambda) - 1) - \frac{1}{\ell(\lambda)} - \varepsilon_\ell(\lambda) \tag{2.544}$$

$$= 1 - \ell'(\lambda) \cdot \varepsilon'(\lambda) - \frac{1}{\ell(\lambda)} - \varepsilon_\ell(\lambda). \tag{2.545}$$

Eq. (2.542) implies now for infinitely many $\lambda \in \mathbb{N}$

$$\begin{aligned} \frac{1}{r(\lambda)} <& 1 - p_\lambda(A(\lambda), x_\lambda) && (2.546) \\ \leq& \ell'(\lambda) \cdot \varepsilon'(\lambda) + \frac{1}{\ell(\lambda)} + \varepsilon_\ell(\lambda) && (2.547) \\ \leq& \frac{1}{2 \cdot r(\lambda)} + \ell'(\lambda) \cdot \varepsilon'(\lambda) + \varepsilon_\ell(\lambda). && (2.548) \end{aligned}$$

However, $\ell' \cdot \varepsilon' + \varepsilon_\ell$ lies in $\mathsf{negl}(t)$. In particular, it cannot be larger than $\frac{1}{2 \cdot r(\lambda)}$ for infinitely many λ, since $r \in \mathsf{poly}(t)$. It follows that we reached a contradiction and that our assumption was false. Hence, the claim of our lemma is true. □

So far, we have been flexible regarding the choice of D. However, the upcoming lemma can only be proven when D is in $O(1)$.

Lemma 107. *Let $D \in O(1)$ and, for each $\lambda \in \mathbb{N}$, let p'_λ be a monotonically decreasing function*

$$p'_\lambda : \{1, \dots, D(\lambda) + 1\} \longrightarrow [0, 1] \qquad (2.549)$$

s.t. $p'_\lambda(1) = 1$ and $p'_\lambda(D(\lambda) + 1) = 0$.

Then, there is an index $i_\dagger$ s.t.

$$\begin{aligned} \forall_\infty \lambda \in \mathbb{N} : \ & i_\dagger \in [D(\lambda)], && (2.550) \\ & 1 - p'_\lambda(i_\dagger) \in \mathsf{negl}(\lambda), && (2.551) \\ & 1 - p'_\lambda(i_\dagger + 1) \notin \mathsf{negl}(\lambda). && (2.552) \end{aligned}$$

Proof. First, we point out that we can assume—without loss of generality—that $D(\lambda)$ is constant and independent of λ. Indeed, if $D(\lambda)$ should alternate infinitely often between two values D_0 and D_1 with $D_0 < D_1$, we can extend p'_λ on the set $[D(\lambda) + 2, D_1 + 1]$ by setting

$$p'_\lambda(i) := 0 \qquad (2.553)$$

for $i > D(\lambda) + 1$. If we now find an index $i_\dagger \in [1, D_1]$ s.t. Eqs. (2.550) to (2.552) are fulfilled with respect to D_1, we know that $i_\dagger$ must be smaller than $D_0 + 1$, since for infinitely many λ we have $p'_\lambda(D_0 + 1) = p'_\lambda(D(\lambda) + 1) = 0$ and $1 - p'_\lambda(i_\dagger)$ must be negligible in λ.

Hence, let D be constant. For $i \in [D + 1]$, consider the function

$$e_i(\lambda) := 1 - p'_\lambda(i). \qquad (2.554)$$

We have $0 = e_1(\lambda) \leq \ldots \leq e_{D+1}(\lambda) = 1$. We can now choose $i_\dagger \in [D]$ maximal s.t. $e_{i_\dagger}(\lambda)$ lies in $\mathsf{negl}(\lambda)$. Since $e_1(\lambda) = 0$ is negligible, such an index $i_\dagger$ must exist. On the other hand, $e_{i_\dagger + 1}(\lambda)$ cannot be negligible, since $i_\dagger$ is maximal and $e_{D+1}(\lambda) = 1$ is non-negligible. □

Unfortunately, Lemma 107 cannot be proven to be true for arbitrary D. In fact, one cannot even hope that there is an index function $i_\dagger : \mathbb{N} \to \mathbb{N}$ and a monotonically increasing function $t' : \mathbb{N} \to \mathbb{N}$ s.t. we have

$$\begin{aligned} & 1 - p'_\lambda(i_\dagger(\lambda)) \in \mathsf{negl}(t'), && (2.555) \\ & 1 - p'_\lambda(i_\dagger(\lambda) + 1) \notin \mathsf{negl}(t'). && (2.556) \end{aligned}$$

Indeed, consider the following counterexample for $D(\lambda) \in \omega(\lambda)$ and any function t'

$$p'_\lambda(i) := \begin{cases} 1, & \text{if } i = 1, \\ 1 - (t'(\lambda))^{i-D(\lambda)-1}, & \text{if } i \in \{2, \ldots, D(\lambda)+1\}. \end{cases} \tag{2.557}$$

Note that we have for each $i > 1$

$$1 - p'_\lambda(i+1) = (t')^{i-D(\lambda)} = t' \cdot (t')^{i-D(\lambda)-1} = t'(1 - p'_\lambda(i)). \tag{2.558}$$

Now, let $i_\dagger : \mathbb{N} \to \mathbb{N}$ s.t.

$$1 - p'_\lambda(i_\dagger(\lambda)) \in \mathsf{negl}(t'), \tag{2.559}$$

$$1 - p'_\lambda(i_\dagger(\lambda)+1) \notin \mathsf{negl}(t'). \tag{2.560}$$

$i_\dagger(\lambda)$ must infinitely often be larger than 1, since $1 - p'_\lambda(2) = (t')^{1-D(\lambda)}$ lies in $\mathsf{negl}(t')$, since $D(\lambda) \in \omega(\lambda)$. However, for $i_\dagger(\lambda) \geq 1$, we have

$$1 - p'_\lambda(i_\dagger(\lambda)+1) = t'(1 - p'_\lambda(i_\dagger(\lambda))) \in \mathsf{negl}(t'). \tag{2.561}$$

Hence, Eqs. (2.559) and (2.560) cannot be fulfilled for $D \in \omega(\lambda)$.

Proof Theorem 99. Let $\mathcal{I} : \mathbb{N} \to \mathcal{P}(\mathbb{N})$ be the map from Lemma 100 s.t. $\mathcal{I}(\lambda) \subseteq [Q(\lambda)]$ and h_{msk} contains non-trivially a monomial $S_{j_1} \cdots S_{j_{D'}}$, $j_1, \ldots, j_{D'} \in \mathcal{I}(\lambda)$, $D' \leq D(\lambda)$, for each good $\mathsf{msk} \in G$. For each $\lambda \in \mathbb{N}$, let

$$j_\lambda(1) < \ldots < j_\lambda(D(\lambda)) \tag{2.562}$$

be an ordering of the elements of $\mathcal{I}(\lambda)$ and set for $i \in [D(\lambda)+1]$

$$R_\lambda(i) := [Q(\lambda)] \setminus \{j_\lambda(1), \ldots, j_\lambda(i-1)\}. \tag{2.563}$$

Note that we have for $i \in [D(\lambda)]$

$$R_\lambda(1) = [Q(\lambda)], \tag{2.564}$$

$$R_\lambda(i+1) = R_\lambda(i) \setminus \{j_\lambda(i)\}, \tag{2.565}$$

$$R_\lambda(D(\lambda)+1) = [Q(\lambda)] \setminus \mathcal{I}(\lambda). \tag{2.566}$$

Now, let $(x_\lambda)_\lambda \in \widetilde{\mathcal{X}}$ be an arbitrary computable sequence of messages. We set for $\lambda \in \mathbb{N}$ and $i \in [D(\lambda)+1]$

$$p'_\lambda(i) := p_\lambda(R_\lambda(i), x_\lambda) = \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda)}} \left[\tau_{\mathsf{msk}, R_\lambda(i)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0\right]. \tag{2.567}$$

We claim that $p'_\lambda(1)$ is always one, $p'_\lambda(D(\lambda)+1)$ is always zero and that p'_λ is monotonically decreasing on $[D(\lambda)+1]$. Indeed, we have

$$p'_\lambda(1) = p_\lambda([Q(\lambda)], x_\lambda) \tag{2.568}$$

$$= \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda)}} [h_{\mathsf{msk}}(\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_Q(\mathsf{ct})) = 0] = 1, \tag{2.569}$$

since $h_{\mathsf{msk}} \circ (\mathsf{sk}_1, \ldots, \mathsf{sk}_Q) = 0$. Further, we have

$$\begin{aligned} p'_\lambda(D(\lambda)+1) =& p_\lambda([Q(\lambda)] \setminus \mathcal{I}(\lambda), x_\lambda) && (2.570) \\ =& \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda)}} \left[\tau_{\mathsf{msk},[Q(\lambda)]\setminus\mathcal{I}(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0\right] = 0, && (2.571) \end{aligned}$$

since h_{msk} must contain a monomial $S_{\iota_1} \cdots S_{\iota_{D'}}$ with $\iota_1, \ldots, \iota_{D'} \in \mathcal{I}(\lambda)$. The map $\tau_{\mathsf{msk},[Q(\lambda)]\setminus\mathcal{I}(\lambda)}$ will not substitute the variables $S_{\iota_1}, \ldots, S_{\iota_{D'}}$ in h_{msk}, hence $\tau_{\mathsf{msk},[Q(\lambda)]\setminus\mathcal{I}(\lambda)}(h_{\mathsf{msk}})(S, C)$ will also contain the monomial $S_{\iota_1} \cdots S_{\iota_{D'}}$ non-trivially. Hence, $\tau_{\mathsf{msk},[Q(\lambda)]\setminus\mathcal{I}(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct})$ cannot be zero. Finally, let $i \in [D(\lambda)]$. We have

$$\begin{aligned} p'_\lambda(i+1) =& p_\lambda(R_\lambda(i+1), x_\lambda) && (2.572) \\ =& \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda)}} \left[\tau_{\mathsf{msk},R_\lambda(i)\setminus\{j_\lambda(i)\}}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0\right] && (2.573) \\ \leq& \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda)}} \left[\tau_{\mathsf{msk},R_\lambda(i)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0\right] = p'_\lambda(i), && (2.574) \end{aligned}$$

since $\tau_{\mathsf{msk},R_\lambda(i)}(h_{\mathsf{msk}})(S, \mathsf{ct})$ results by substitution of the variable $S_{j_\lambda(i)}$ in the polynomial $\tau_{\mathsf{msk},R_\lambda(i)\setminus\{j_\lambda(i)\}}(h_{\mathsf{msk}})(S, \mathsf{ct})$ by $\mathsf{sk}_{j_\lambda(i)}(\mathsf{ct})$. Hence, if the polynomial $\tau_{\mathsf{msk},R_\lambda(i)\setminus\{j_\lambda(i)\}}(h_{\mathsf{msk}})(S, \mathsf{ct})$ vanishes, then $\tau_{\mathsf{msk},R_\lambda(i)}(h_{\mathsf{msk}})(S, \mathsf{ct})$ must vanish, too.

According to Lemma 107, there must exist a constant $i_\dagger \in \mathbb{N}$ s.t. we have:

$$\begin{aligned} &\forall_\infty \lambda \in \mathbb{N}: \quad i_\dagger \in [D(\lambda)], \\ &1 - p'_\lambda(i_\dagger) = 1 - \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda)}} \left[\tau_{\mathsf{msk},R_\lambda(i_\dagger)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0\right] \in \mathsf{negl}(\lambda), \\ &1 - p'_\lambda(i_\dagger + 1) = 1 - \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda)}} \left[\tau_{\mathsf{msk},R_\lambda(i_\dagger+1)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0\right] \notin \mathsf{negl}(\lambda). \end{aligned}$$

The claim of Theorem 99 now follows by setting

$$\begin{aligned} B(\lambda) :=& R_\lambda(i_\dagger), \\ A(\lambda) :=& R_\lambda(i_\dagger + 1) = B(\lambda) \setminus \{j_\lambda(i_\dagger)\}. \end{aligned} \qquad \square$$

2.5.4 Compact Functional Encryption

We close this chapter by proving lower bounds for *compact* lattice-based FE schemes. Concretely, we will show that the size of ciphertexts of lattice-based FE schemes for polynomial functions cannot be smaller than certain thresholds that depend on the degree of secret keys. Towards this end, we introduce the space of e-linear functions over $\mathbb{Z}_p$. We call a function $f\colon (\mathbb{Z}_p^n)^e \to \mathbb{Z}_p$ ***e*-linear** iff, for vectors of variables $X^{(1)} = (X_1^{(1)}, \ldots, X_n^{(1)}), \ldots, X^{(e)} = (X_1^{(e)}, \ldots, X_n^{(e)})$, the expression $f(X^{(1)}, \ldots, X^{(e)})$ is linear in $X^{(i)}$ for each $i \in [e]$. Equivalently, one can require that $f(X^{(1)}, \ldots, X^{(e)})$ is given by $\overline{f}(X^{(1)} \otimes \ldots \otimes X^{(e)})$, for a linear function $\overline{f} : \mathbb{Z}_p^{n^e} \to \mathbb{Z}_p$, where $\otimes$ denotes the Kronecker product. Note that the space of e-linear functions subsumes the space of degree-e polynomials in the following way: for each degree-e polynomial $f \in \mathbb{Z}_p[Y_1, \ldots, Y_n]$ exists an e-linear function $\widetilde{f}$ over $e \cdot n$ variables s.t. we have $f(x) = \widetilde{f}(x, \ldots, x)$ for each $x \in \mathbb{Z}_p^n$.

In the following, we consider the functionality $\mathcal{F}\colon \mathcal{X} \to \mathcal{Y}$ of e-linear functions where the message space is $\mathcal{X} = \mathbb{Z}_p^{e \times n} = \mathbb{Z}_p^n \times \ldots \times \mathbb{Z}_p^n$ and the value space is $\mathcal{Y} = \mathbb{Z}_p$.

Theorem 108 (Lower Bounds for Compact FE)**.** *Let* $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ *be an FE scheme for the functionality of e-linear functions $\mathcal{F} : \mathbb{Z}_p^{e \times n} \to \mathbb{Z}_p$. Assume that* FE *is lattice-based with encryption depth d_1, decryption depth d_2 and noise-bound B and that ciphertexts of* FE *lie in $\mathbb{Z}_q^m$.*

Let $d_2 > 1$ and $q > p > 2$ with q prime. Let $m \geq \lambda$ s.t.

$$m \in O(n^{e/d_2}). \tag{2.575}$$

There is a constant $D \in \mathbb{N}$ s.t. whenever FE *is correct, and we have*

$$\max(2d, D+1) \leq p/2, \qquad \text{and} \qquad 2 \cdot (d+1)! \cdot (2d)! \cdot \Gamma_d \cdot B < q, \tag{2.576}$$

for $d := d_1 \cdot d_2$, then there is an adversary $\mathcal{A}$ against the selective IND-CPA security of FE *that makes* $\mathsf{poly}(B+\lambda)$ *encryption queries, $O(n^e)$ function queries and* $\mathsf{poly}(B+\lambda)$ *arithmetic operations over $\mathbb{Z}_p$, $\mathbb{Z}_q$ and $\mathbb{Z}$. The advantage of $\mathcal{A}$ in Game 3 is noticeable in λ.*

Let us explain the parameter restrictions of Theorem 108. In Theorem 108, we demand that we have

$$m \in O(n^{e/d_2}), \tag{2.577}$$

where $d_2 > 1$ is the degree of secret keys of FE. If we consider for the example the functionality of multilinear functions of degree $e = 2$, then Theorem 108 implies that there is no lattice-based FE scheme, in which ciphertexts are of linear size $m \in \Theta(n)$ and secret keys are of degree $d_2 = 2$. However, note that Theorem 108 cannot exclude the existence of FE schemes where ciphertexts are of quadratic size $\Theta(n^2)$ and secret keys are of degree 1, since such schemes can be bootstrapped by relinearizing quadratic functions and using a secure inner-product encryption scheme. If we consider the functionality of functions of degree $e = 4$, then Theorem 108 excludes the existence of FE schemes where ciphertexts are linearly compact and secret keys are of degree 4, and of FE schemes where ciphertexts are of quadratic size $\Theta(n^2)$ and secret keys are of degree $d_2 = 2$.

In general, Theorem 108 states that whenever we have a secure lattice-based FE scheme for polynomials of degree $e > 1$ where ciphertexts are of some compact size $m \in O(n^{e-a})$, for some constant $a > 0$, then the secret keys of this scheme must have a degree of at least $d_2 > e/(e-a)$. Now, this result does not fully exclude the existence of secure lattice-based FE schemes for polynomials of degree $e > 1$, since we can always increase the degree d_2 of secret keys to circumvent the claim of Theorem 108. I suspect that this problem can be fixed with regard to subexponential adversaries. In fact, if one could prove Conjecture 2, one could adapt the proof of this subsection to show the following more general version of Theorem 108:

Conjecture 3. *Let* $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ *be an FE scheme for the functionality of e-linear functions. Assume that* FE *is lattice-based with noise-bound* B *and decryption depth* d_2 *and that ciphertexts of* FE *lie in* $\mathbb{Z}_q^m$ *with* q *prime. Assume further that* $p \in \omega(1)$*,* $q \in \omega(B)$ *and*

$$m \in O(n^{e-a}) \tag{2.578}$$

for some constant $a > 0$*.*

There is a $D \in O\big((m^{d_2}/n^e)^{1/(d_2-1)}\big) = O(n^{e-a \cdot d_2/(d_2-1)})$ *and an adversary* $\mathcal{A}$ *against the selective IND-CPA security of* FE *that makes* $\mathsf{poly}(B+\lambda^D)$ *encryption queries,* $O(n^e)$ *function queries and* $\mathsf{poly}(B + \lambda^D)$ *arithmetic operations. The advantage of* $\mathcal{A}$ *in Game 3 is noticeable in* λ^D*.*

We will pursue the same strategy as in Section 2.5.2 and use Theorem 93 to disprove the security of FE. For this end, let us first define a partial SKE scheme $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \_)$ by giving a preceding setup algorithm $\mathsf{Setup}'_{\mathsf{Pre}}$:

Algorithm 17. Set $Q := n^e$ and let

$$D' \in O((m^{d_2}/Q)^{1/(d_2-1)}) = O(((n^{e/d_2})^{d_2}/n^e)^{1/(d_2-1)}) = O(1) \tag{2.579}$$

be the function from Theorem 10 for Q polynomials over m variables of degree d_2. Set

$$D := \max_{\lambda \in \mathbb{N}} D(\lambda) \tag{2.580}$$

and $M := \max(D + 1, 2 \cdot d)$.

Let $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \_)$ be the partial SKE from Algorithm 14 for the message space $\mathcal{X}' = \{0, \dots, M\} \subset \mathbb{Z}_p$ whose setup algorithm is based on the following preceding setup algorithm $\mathsf{Setup}'_{\mathsf{Pre}}$:

$\mathsf{Setup}'_{\mathsf{Pre}}$: On input 1^λ, $\mathsf{Setup}'_{\mathsf{Pre}}$ computes—in a deterministic way—an enumeration

$$\beta_1 = (\beta_{1,1}, \dots, \beta_{1,e}), \dots, \beta_Q = (\beta_{Q,1}, \dots, \beta_{Q,e}) \tag{2.581}$$

of all elements in $[n]^e$. For $i \in [n^e] = [Q]$, it sets

$$f_i(X^{(1)}, \dots, X^{(e)}) := X^{(1)}_{\beta_{i,1}} \cdots X^{(e)}_{\beta_{i,e}}. \tag{2.582}$$

It samples an index $i_* \leftarrow [Q]$ uniformly at random and denotes by ν the affine linear map

$$\nu : \mathbb{Z}_p \longrightarrow \mathbb{Z}_p^n \times \ldots \times \mathbb{Z}_p^n \tag{2.583}$$

$$x \longmapsto (x \cdot u_{\beta_{i_*,1}}, u_{\beta_{i_*,2}}, \dots, u_{\beta_{i_*,e}}), \tag{2.584}$$

where $u_i \in \mathbb{Z}_p^n$ denotes the i-th unit vector for $i \in [n]$.

$\mathsf{Setup}'_{\mathsf{Pre}}$ outputs

$$(f_1, \dots, f_{i_*-1}, f_{i_*+1}, \dots, f_Q, f_{i_*}, \nu). \tag{2.585}$$

Let us informally explain what $\mathsf{Setup}'_{\mathsf{Pre}}$ does: it enumerates deterministically all monomials $f_1, \dots, f_Q$ of $\mathcal{F}$. Then, it selects randomly one monomial f_{i_*}. This monomial will become our attack point, later. $\mathsf{Setup}'_{\mathsf{Pre}}$ computes an affine map $\nu : \mathbb{Z}_p \to \mathbb{Z}_p^n \times \ldots \times \mathbb{Z}_p^n$ s.t. $\nu(x)$ is non-zero at the positions that match the monomials of f_{i_*}. At one of these positions, ν plants the value x, at all other positions, it assigns the value 1. Hence, we have for each $i \in [Q]$

$$f_i(\nu(x)) = \begin{cases} x, & \text{if } i = i_*, \\ 0, & \text{if } i \neq i_*. \end{cases} \tag{2.586}$$

To construct Dec', we first derandomize the key generation algorithm KeyGen of FE. Concretely, we will show that an adversary on the selective IND-CPA security of FE can assume—without loss of generality—that KeyGen is a deterministic algorithm, as long as the adversary queries for each function at most one secret key.

Lemma 109. *Let $\mathcal{A}$ be an adversary in the selective IND-CPA Game 3 of* FE *that, for each function $f \in \mathcal{F}$, queries at most one secret key $\mathsf{sk}_f \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f)$.*

When analysing the advantage of $\mathcal{A}$ in Game 3 we can assume—without loss of generality—that KeyGen *is a deterministic algorithm.*

Proof. We will construct an alternative FE scheme $\widetilde{\mathsf{FE}} = (\widetilde{\mathsf{Setup}}, \widetilde{\mathsf{KeyGen}}, \widetilde{\mathsf{Enc}}, \widetilde{\mathsf{Dec}})$ from $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ with a deterministic key generation algorithm s.t. the view of $\mathcal{A}$ in Game 3 with FE is identical to $\mathcal{A}$'s view in Game 3 with $\widetilde{\mathsf{FE}}$.

The scheme $\widetilde{\mathsf{FE}} = (\widetilde{\mathsf{Setup}}, \widetilde{\mathsf{KeyGen}}, \widetilde{\mathsf{Enc}}, \widetilde{\mathsf{Dec}})$ is given as follows:

$\widetilde{\mathsf{Setup}}$: On input 1^λ, $\widetilde{\mathsf{Setup}}$ computes $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$. Additionally, for each $f \in \mathcal{F}_\lambda$, it samples $\mathsf{sk}_f \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f)$ and outputs

$$\widetilde{\mathsf{msk}} := (\mathsf{msk}, (\mathsf{sk}_f)_{f \in \mathcal{F}_\lambda}). \tag{2.587}$$

$\widetilde{\mathsf{KeyGen}}$: On input $\widetilde{\mathsf{msk}} = (\mathsf{msk}, (\mathsf{sk}_f)_{f \in \mathcal{F}_\lambda})$ and $g \in \mathcal{F}_\lambda$, $\widetilde{\mathsf{KeyGen}}$ outputs sk_f.

$\widetilde{\mathsf{Enc}}$: On input $\widetilde{\mathsf{msk}} = (\mathsf{msk}, (\mathsf{sk}_f)_{f \in \mathcal{F}_\lambda})$ and $x \in \mathcal{X}_\lambda$, $\widetilde{\mathsf{Enc}}$ computes and outputs

$$\mathsf{ct}_x \leftarrow \mathsf{Enc}(\mathsf{msk}, x). \tag{2.588}$$

$\widetilde{\mathsf{Dec}}$: On input sk and ct, $\widetilde{\mathsf{Dec}}$ computes and outputs

$$y \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{ct}). \tag{2.589}$$

Note that the size of $\widetilde{\mathsf{msk}}$ is at least exponential in λ, since $\widetilde{\mathsf{msk}}$ needs to contain a secret key for each function $f \in \mathcal{F}_\lambda$. However, this is fine, since we do not require in this work any bounds on the size of msk or on the time complexity of Setup. Indeed, the advantages of the adversaries from Theorem 93 and Algorithm 16 are independent of the time complexity of Setup and KeyGen.

Now, let $\mathcal{A}$ be any adversary for the selective IND-CPA Game 3 of FE. If $\mathcal{A}$ requests secret keys for Q different functions $f_1, \ldots, f_Q \in \mathcal{F}_\lambda$, then the distributions $(\mathsf{sk}_i)_{i=1}^Q$, $\mathsf{sk}_i \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_i)$, and $(\widetilde{\mathsf{sk}}_i)_{i=1}^Q$, $\widetilde{\mathsf{sk}}_i \leftarrow \widetilde{\mathsf{KeyGen}}(\mathsf{msk}, f_i)$, are identical. Hence, the view of $\mathcal{A}$ in Game 3 for FE is identical to the view of $\mathcal{A}$ in Game 3 for $\widetilde{\mathsf{FE}}$. □

Now, since we can assume that KeyGen is deterministic, and since the collection of functions $f_1, \ldots, f_Q \in \mathcal{F}_\lambda$ is deterministically computed by $\mathsf{Setup}'_{\mathsf{Pre}}(1^\lambda)$, we can follow our argument at the beginning of Section 2.5.3 and see that for each master secret key msk there exists a unique polynomial $h_{\mathsf{msk}} \in \mathbb{Z}_q[S_1, \ldots, S_Q]$ s.t.

$$h_{\mathsf{msk}} \neq 0, \tag{2.590}$$

$$\deg h_{\mathsf{msk}} \leq D, \tag{2.591}$$

$$h(\mathsf{sk}_1, \ldots, \mathsf{sk}_Q) = 0 \in \mathbb{Z}_q[C_1, \ldots, C_m]. \tag{2.592}$$

Proposition 110. *Let $u_i \in \mathbb{Z}_p^n$ be the i-th unit vector, i.e.*

$$u_i = (0, \ldots, 0, 1, 0, \ldots, 0) \tag{2.593}$$

has a one at the i-th position and is zero everywhere else.

Consider the following subspace $\widetilde{\mathcal{X}} = (\widetilde{\mathcal{X}}_\lambda)_\lambda$ of the message space $\mathcal{X} = (\mathbb{Z}_p^n)^e = \mathbb{Z}_p^n \times \ldots \times \mathbb{Z}_p^n$

$$\widetilde{\mathcal{X}}_\lambda := \{(x \cdot u_{i_1}, u_{i_2}, \ldots, u_{i_e}) \mid i_1, \ldots, i_e \in [n], x \in \{0, \ldots, M\}\}. \tag{2.594}$$

The size of $\widetilde{\mathcal{X}}_\lambda$ is given by

$$\#\widetilde{\mathcal{X}}_\lambda = (M \cdot n + 1) \cdot n^{e-1} \in \mathsf{poly}(\lambda) \tag{2.595}$$

and for every output $(f_1, \ldots, f_{i_*-1}, f_{i_*+1}, \ldots, f_Q, f_{i_*}, \nu)$ of $\mathsf{Setup}'_{\mathsf{Pre}}(1^\lambda)$ from Algorithm 17 and every value $x \in \{0, \ldots, M\}$, we have

$$\nu(x) \in \widetilde{\mathcal{X}}_\lambda. \tag{2.596}$$

Theorem 99 postulates that there is a set G of master secret keys, maps $A, B : \mathbb{N} \to \mathcal{P}(\mathbb{N})$ and functions $\varepsilon \in \mathsf{negl}(\lambda)$, $\rho \notin \mathsf{negl}(\lambda)$ s.t. we have:

1. $A(\lambda) \subset B(\lambda) \subseteq [Q(\lambda)]$ and $\#A(\lambda) = \#B(\lambda) - 1$,
2. $\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)}[\mathsf{msk} \in G] \geq Q^{-D}$,
3. for all $x \in \widetilde{\mathcal{X}}_\lambda$

$$\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)}} \left[\tau_{\mathsf{msk}, B(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0 \mid \mathsf{msk} \in G\right] \geq 1 - \varepsilon, \tag{2.597}$$

$$\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)}} \left[\tau_{\mathsf{msk}, A(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0 \mid \mathsf{msk} \in G\right] \leq 1 - \rho. \tag{2.598}$$

We can now give a formal description of the decryption algorithm Dec' for that we will show a non-negligible advantage at decrypting ciphertexts of $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \mathsf{Dec}')$:

Algorithm 18. On input a master secret key

$$\mathsf{msk}' = (\mathsf{msk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_{i_*-1}, \mathsf{sk}_{i_*+1}, \ldots, \mathsf{sk}_Q, \mathsf{sk}_{i_*}, \nu) \tag{2.599}$$

(where $\mathsf{sk}_i = \mathsf{KeyGen}(\mathsf{msk}, f_i)$ for $i \in [Q]$), and a ciphertext

$$\mathsf{ct}' = (\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_{i_*-1}(\mathsf{ct}), \mathsf{sk}_{i_*+1}(\mathsf{ct}), \ldots, \mathsf{sk}_Q(\mathsf{ct})) \in \mathbb{Z}_q^{Q-1} \tag{2.600}$$

(where $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, \nu(x))$ for $x \in \{0, \ldots, M\}$), Dec' proceeds in the following steps:

Step 1: Let G, $A(\lambda)$, $B(\lambda)$ be the sets from Theorem 99. Dec' checks if msk lies in G. If msk does not lie in G, Dec' terminates by outputting a uniformly random element of $\mathcal{X}'_\lambda = \{0, \dots, M\}$.

Step 2: Let $i_\dagger \in [Q(\lambda)]$ be s.t.

$$\{i_\dagger\} = B(\lambda) \setminus A(\lambda). \tag{2.601}$$

Since the enumeration $f_1, \dots, f_Q$ computed by $\mathsf{Setup}'_{\mathsf{Pre}}(1^\lambda)$ is deterministic, Dec' can determine the index $i_* \in [Q(\lambda)]$ sampled by $\mathsf{Setup}'_{\mathsf{Pre}}(1^\lambda)$ for computing msk'. If $i_\dagger \neq i_*$, Dec' terminates by outputting a uniformly random element of $\mathcal{X}'_\lambda$.

Step 3: Let $(c_1, \dots, c_{i_\dagger - 1}, c_{i_\dagger + 1}, \dots, c) := \mathsf{ct}'$ be the coordinates of ct'. Let

$$\zeta : \mathbb{Z}_q[S_1, \dots, S_Q] \longrightarrow \mathbb{Z}_q[S_1, \dots, S_Q] \tag{2.602}$$

$$c_i \longmapsto \begin{cases} S_i, & \text{if } i \notin A(\lambda), \\ c_i, & \text{if } i \in A(\lambda) \end{cases} \tag{2.603}$$

be the ring morphism that substitutes S_i by c_i if $i \in A(\lambda)$. Dec' computes

$$\zeta(h_{\mathsf{msk}}) = h_{\mathsf{msk}}((S_i)_{i \notin A(\lambda)}, (c_i)_{i \in A(\lambda)}). \tag{2.604}$$

If $\zeta(h_{\mathsf{msk}}) = 0$, then Dec' terminates by outputting a uniformly random element of $\mathcal{X}'_\lambda$.

Step 4: Dec' constructs the following sets

$$U := \{u \in \mathbb{Z}_q^m \mid \tau_{\mathsf{msk}, B(\lambda)}(h_{\mathsf{msk}})(S, u) = 0\}, \tag{2.605}$$

$$V := \{u \in U \mid \forall i \in [Q(\lambda)] \setminus \{i_\dagger\} : \ \mathsf{sk}_i(u) = c_i\}, \tag{2.606}$$

$$W := \{\mathsf{sk}_{i_\dagger}(u) \mid u \in U\}. \tag{2.607}$$

Step 5: If W is empty, then Dec' terminates by outputting a uniformly random element of $\mathcal{X}'_\lambda$. Otherwise, it draws $w \leftarrow W$ and assumes that w is of the shape $w = \mathsf{sk}_{i_\dagger}(\mathsf{ct})$. It uses w to compute and output $\mathsf{Dec}(\mathsf{sk}_{i_\dagger}, \mathsf{ct})$.

Remark 10 (Uncomputability of Dec'). Note that Dec' does not need to be computable. Indeed, if Setup and KeyGen of FE are not computable, then the sets G, $A(\lambda)$ and $B(\lambda)$ of Theorem 99 may be incomputable. However, Dec' does not need to be computable, since it is not a part of the adversaries on the IND-CPA security of FE. Indeed, we only need Dec' as a mathematical function to prove that the statistical distance between ciphertexts of SKE' for different messages is noticeable.

Lemma 111. *Let* $x \in \mathcal{X}'_\lambda$. *For* $\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda)$ *and* $\mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x)$, *denote by* terminate *the event that Step 5 is* not *reached by* $\mathsf{Dec}'(\mathsf{msk}', \mathsf{ct}')$.

If FE *is secure against the adversary from Algorithm 16 (for each* $\ell \in \mathsf{poly}(\lambda)$*), we have*

$$\Pr_{\substack{\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda) \\ \mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x)}}[\neg \mathsf{terminate}] \geq \frac{\rho}{Q^{D+1}} \notin \mathsf{negl}(\lambda). \tag{2.608}$$

Proof. Let $x \in \mathcal{X}'_\lambda$ and sample $\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda)$. Note that msk' must be of the shape

$$\mathsf{msk}' = (\mathsf{msk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_{i_*-1}, \mathsf{sk}_{i_*+1}, \ldots, \mathsf{sk}_Q, \mathsf{sk}_{i_*}, \nu) \leftarrow \mathsf{Setup}'(1^\lambda) \tag{2.609}$$

where

$$\begin{align} &\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda), \tag{2.610}\\ &(f_1, \ldots, f_{i_*-1}, f_{i_*+1}, \ldots, f_Q, f_{i_*}, \nu) \leftarrow \mathsf{Setup}'_{\mathsf{Pre}}(1^\lambda), \tag{2.611}\\ &\forall i \in [Q] : \quad \mathsf{sk}_i = \mathsf{KeyGen}(\mathsf{msk}, f_i). \tag{2.612} \end{align}$$

Further, sample $\mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x)$ and note that ct' must be of the shape

$$\mathsf{ct}' = (\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_{i_*-1}(\mathsf{ct}), \mathsf{sk}_{i_*+1}(\mathsf{ct}), \ldots, \mathsf{sk}_Q(\mathsf{ct})) \in \mathbb{Z}_q^{Q-1} \tag{2.613}$$

for

$$\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, \nu(x)). \tag{2.614}$$

Note that $\mathsf{terminate}$ will occur exactly iff one of the following conditions does not hold:

$$\begin{align} \mathsf{msk} &\in G, \tag{2.615}\\ i_\dagger &= i_*, \tag{2.616}\\ \zeta(h_{\mathsf{msk}}) &\neq 0. \tag{2.617} \end{align}$$

Because of Theorem 99, the value $i_\dagger$ is fixed for each λ. The value $i_* \leftarrow [Q]$ is drawn uniformly at random, hence

$$\Pr[i_* = i_\dagger] = \frac{1}{Q}. \tag{2.618}$$

Further, Theorem 99 implies

$$\Pr[\mathsf{msk} \in G] \geq \frac{1}{Q^D}. \tag{2.619}$$

Note that both events are independent of each other. Now, we want to lower bound $\Pr[\zeta(h_{\mathsf{msk}}) \neq 0]$ conditioned on $\mathsf{msk} \in G$. ζ replaces each occurrence of S_i in h_{msk} by $\mathsf{sk}_i(\mathsf{ct})$ if $i \in A(\lambda)$. Hence, we have

$$\zeta(h_{\mathsf{msk}}) = \tau_{\mathsf{msk}, A(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct}). \tag{2.620}$$

Because of Theorem 99, the probability that $\tau_{\mathsf{msk}, A(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct})$ is not zero if $\mathsf{msk} \in G$ is bounded by

$$\Pr[\tau_{\mathsf{msk}, A(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct}) \mid \mathsf{msk} \in G] \geq \rho(\lambda) \tag{2.621}$$

for the noticeable function ρ. Hence, we have

$$\begin{align} \Pr[\neg\mathsf{terminate}] =& \Pr[i_* = i_\dagger] \cdot \Pr[\mathsf{msk} \in G] \cdot \Pr[\zeta(h_{\mathsf{msk}})(\mathsf{ct}) \mid \mathsf{msk} \in G] \tag{2.622}\\ \geq& \frac{1}{Q} \cdot \frac{1}{Q^D} \cdot \rho = \frac{\rho}{Q^{D+1}} \notin \mathsf{negl}(\lambda). \end{align}$$

□

Lemma 112. *Let* $x \in \mathcal{X}'_\lambda$, $\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda)$ *and* $\mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x)$.

Denote by vanish *the event that*

$$\tau_{\mathsf{msk},B(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0. \tag{2.623}$$

If FE *is secure against the adversary from Algorithm 16, we have*

$$\Pr_{\substack{\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda) \\ \mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x)}} [\mathsf{vanish} \mid \neg \mathsf{terminate}] \geq 1 - \frac{\varepsilon(\lambda)}{\rho(\lambda)} \tag{2.624}$$

whenever $\rho(\lambda) > 0$.

Proof. Sample

$$\mathsf{msk}' = (\mathsf{msk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_{i_*-1}, \mathsf{sk}_{i_*+1}, \ldots, \mathsf{sk}_Q, \mathsf{sk}_{i_*}, \nu) \leftarrow \mathsf{Setup}'(1^\lambda), \tag{2.625}$$

$$\mathsf{ct}' = (\mathsf{sk}_1(\mathsf{ct}), \ldots, \mathsf{sk}_{i_*-1}(\mathsf{ct}), \mathsf{sk}_{i_*+1}(\mathsf{ct}), \ldots, \mathsf{sk}_Q(\mathsf{ct})) \leftarrow \mathsf{Enc}'(\mathsf{msk}', x) \tag{2.626}$$

and assume that terminate does not occur in $\mathsf{Dec}'(\mathsf{msk}', \mathsf{ct}')$.

In this case, we have

$$\mathsf{msk} \in G, \tag{2.627}$$

$$i_* = i_\dagger, \tag{2.628}$$

$$\tau_{\mathsf{msk},A(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct}) \neq 0. \tag{2.629}$$

Theorem 99 implies that we have

$$\Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x)}} \left[\tau_{\mathsf{msk},B(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0 \mid \mathsf{msk} \in G\right] \geq 1 - \varepsilon. \tag{2.630}$$

However, while the event $\tau_{\mathsf{msk},B(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0$ is independent of Eq. (2.628), it may be correlated with Eq. (2.629). Denote the event $\tau_{\mathsf{msk},A(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0$ by T_A and the event $\tau_{\mathsf{msk},B(\lambda)}(h_{\mathsf{msk}})(S, \mathsf{ct}) = 0$ by T_B and denote by $\mathsf{Setup}_{\mathsf{good}}(1^\lambda)$ the output distribution of $\mathsf{Setup}(1^\lambda)$ restricted to G. We then have

$$\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda)} [T_B \mid \neg T_A] \tag{2.631}$$

$$= \frac{\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda)} [T_B \wedge \neg T_A]}{\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda)} [\neg T_A]} \tag{2.632}$$

$$\geq \frac{\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda)} [T_B] + \Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda)} [\neg T_A] - 1}{\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda)} [\neg T_A]} \tag{2.633}$$

$$\geq \frac{1 - \varepsilon + \Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda)} [\neg T_A] - 1}{\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda)} [\neg T_A]} \tag{2.634}$$

$$= \frac{\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda)} [\neg T_A] - \varepsilon}{\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda)} [\neg T_A]} \tag{2.635}$$

$$= 1 - \frac{\varepsilon}{\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}_{\mathsf{good}}(1^\lambda)} [\neg T_A]} \geq 1 - \frac{\varepsilon}{\rho}. \tag{2.636}$$

Hence, we have

$$\Pr_{\substack{\mathsf{msk}\leftarrow\mathsf{Setup}(1^\lambda)\\ \mathsf{ct}\leftarrow\mathsf{Enc}(\mathsf{msk},x)}}\left[\tau_{\mathsf{msk},B(\lambda)}(h_{\mathsf{msk}})(S,\mathsf{ct})=0 \mid \neg\mathsf{terminate}\right] \geq 1-\varepsilon/\rho. \qquad \square$$

Lemma 113. *For $x \in \mathcal{X}'_\lambda$, we have*

$$\Pr_{\substack{\mathsf{msk}'\leftarrow\mathsf{Setup}'(1^\lambda)\\ \mathsf{ct}'\leftarrow\mathsf{Enc}'(\mathsf{msk}',x)}}[\mathsf{Dec}'(\mathsf{msk}',\mathsf{ct}')=x \mid \mathsf{vanish},\neg\mathsf{terminate}] \geq \frac{\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})}{D} \tag{2.637}$$

Proof. Assume that $\neg\mathsf{terminate}$ and vanish do hold. In this case, we have

$$i_* = i_\dagger, \tag{2.638}$$

$$\tau_{\mathsf{msk},B(\lambda)}(h_{\mathsf{msk}})(S,\mathsf{ct}) = 0, \tag{2.639}$$

$$\tau_{\mathsf{msk},A(\lambda)}(h_{\mathsf{msk}})(S,\mathsf{ct}) \neq 0. \tag{2.640}$$

Let us interpret $\tau_{\mathsf{msk},A(\lambda)}(h_{\mathsf{msk}})(S,\mathsf{ct})$ as a univariate polynomial in the variable $S_{i_\dagger}$ with coefficients in $\mathbb{Z}_q[S_1,\dots,S_{i_\dagger-1},S_{i_\dagger+1},\dots,S_Q]$ and let us call this polynomial

$$g(S_{i_\dagger}) := \tau_{\mathsf{msk},A(\lambda)}(h_{\mathsf{msk}})(S,\mathsf{ct}) \in (\mathbb{Z}_q[S_1,\dots,S_{i_\dagger-1},S_{i_\dagger+1},\dots,S_Q])[S_{i_\dagger}].$$

Since $B(\lambda) = A(\lambda) \cup \{i_\dagger\}$ and because of Eqs. (2.639) and (2.640), we have

$$g(\mathsf{sk}_{i_\dagger}(\mathsf{ct})) = \tau_{\mathsf{msk},B(\lambda)}(h_{\mathsf{msk}})(S,\mathsf{ct}) = 0, \tag{2.641}$$

$$g(S_{i_\dagger}) = \tau_{\mathsf{msk},A(\lambda)}(h_{\mathsf{msk}})(S,\mathsf{ct}) \neq 0. \tag{2.642}$$

Hence, g is a non-zero polynomial whose set of roots contains $\mathsf{sk}_{i_\dagger}(\mathsf{ct})$. Since $\deg g \leq \deg h_{\mathsf{msk}} \leq D$, the set of roots of g contains at most D elements in $\mathbb{Z}_q$.

Now, consider the sets

$$U := \left\{u \in \mathbb{Z}_q^m \;\middle|\; \tau_{\mathsf{msk},B(\lambda)}(h_{\mathsf{msk}})(S,u) = 0\right\}, \tag{2.643}$$

$$V := \{u \in U \mid \forall i \in [Q(\lambda)] \setminus \{i_\dagger\}:\ \mathsf{sk}_i(u) = \mathsf{sk}_i(\mathsf{ct})\}, \tag{2.644}$$

$$W := \left\{\mathsf{sk}_{i_\dagger}(u) \;\middle|\; u \in U\right\} \tag{2.645}$$

computed by $\mathsf{Dec}'(\mathsf{msk}',\mathsf{ct}')$ in Step 4. Note that ct must lie in U and V, since vanish does hold. We claim that W is contained in the set of roots of g. In fact, let $\mathsf{sk}_{i_\dagger}(u) \in W$ for some $u \in V$. We have

$$g(\mathsf{sk}_{i_\dagger}(u)) = \tau_{\mathsf{msk},A(\lambda)}(h_{\mathsf{msk}})(S_1,\dots,S_{i_\dagger-1},\mathsf{sk}_{i_\dagger}(u),S_{i_\dagger+1},\dots,S_Q,\mathsf{ct}) \tag{2.646}$$

$$= h_{\mathsf{msk}}(T_1,\dots,T_Q) \tag{2.647}$$

where we have for $i \in [Q(\lambda)]$

$$T_i = \begin{cases} \mathsf{sk}_i(\mathsf{ct}), & \text{if } i \in A(\lambda),\\ \mathsf{sk}_{i_\dagger}(u), & \text{if } i = i_\dagger,\\ S_i, & \text{otherwise.}\end{cases} \tag{2.648}$$

Since u lies in V, we have $\mathsf{sk}_i(u) = \mathsf{sk}_i(\mathsf{ct})$ for each $i \in A(\lambda)$. Hence, we can simplify T_i to

$$T_i = \begin{cases} \mathsf{sk}_i(u), & \text{if } i \in B(\lambda),\\ S_i, & \text{otherwise,}\end{cases} \tag{2.649}$$

for $i \in [Q(\lambda)]$. Since $u \in U$, it follows

$$g(\mathsf{sk}_{i_\dagger}(u)) = h_{\mathsf{msk}}(T_1, \ldots, T_Q) = \tau_{\mathsf{msk},B(\lambda)}(h_{\mathsf{msk}})(S, u) = 0. \tag{2.650}$$

Hence, W is contained in the set of roots of g, which contains at most D elements. Since $\mathsf{ct} \in V$, W must contain $\mathsf{sk}_{i_\dagger}(\mathsf{ct})$.

It follows that in Step 5, the probability that $\mathsf{Dec}'(\mathsf{msk}', \mathsf{ct}')$ draws $w = \mathsf{sk}_{i_\dagger}(\mathsf{ct})$ from W is at least $1/D$. If $\mathsf{Dec}'(\mathsf{msk}', \mathsf{ct}')$ draws this element and uses Dec to decrypt, its output is given by $\mathsf{Dec}(\mathsf{sk}_{i_\dagger}, \mathsf{ct})$. Hence, we have

$$\Pr_{\substack{\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda) \\ \mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x)}} [\mathsf{Dec}'(\mathsf{msk}', \mathsf{ct}') = x \mid \mathsf{vanish}, \neg\mathsf{terminate}] \tag{2.651}$$

$$\geq \frac{1}{D} \cdot \Pr_{\substack{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, \nu(x))}} \left[\mathsf{Dec}(\mathsf{sk}_{i_\dagger}, \mathsf{ct}) = f_{i_\dagger}(\nu(x))\right] \geq \frac{\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})}{D}, \tag{2.652}$$

since $f_{i_\dagger}(\nu(x)) = x$. □

Lemma 114. *If* FE *is correct and secure against the adversary from Algorithm 16, then* Dec' *has a non-negligible advantage at decrypting messages of* SKE'*. Concretely, we have*

$$\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') \geq \frac{M+1}{M} \cdot \left(\frac{\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})}{D} \cdot \left(1 - \frac{\varepsilon(\lambda)}{\rho(\lambda)}\right) - \frac{1}{M+1} \right) \cdot \frac{\rho}{Q^{D+1}}$$

whenever

$$\frac{\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})}{D} \cdot \left(1 - \frac{\varepsilon(\lambda)}{\rho(\lambda)}\right) \geq \frac{1}{M+1}. \tag{2.653}$$

Proof. Note that Eq. (2.653) must hold for infinitely many $\lambda \in \mathbb{N}$, since $\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})$ is overwhelming, $M \geq D$, ε is negligible and ρ is noticeable. Now, let $\lambda \in \mathbb{N}$ s.t. it does hold.

Draw $\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda)$ and $\mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x)$ for some $x \in \{0, \ldots, M\}$, and denote by $\mathsf{correct}$ the event that

$$\mathsf{Dec}'(\mathsf{msk}', \mathsf{ct}') = x. \tag{2.654}$$

Note that if $\mathsf{terminate}$ does occur, then Dec' terminated in Steps 1 to 3. In this case, we have

$$\Pr[\mathsf{correct} \mid \mathsf{terminate}] = \frac{1}{M+1}. \tag{2.655}$$

Otherwise, we have

$$\begin{aligned}
&\Pr[\mathsf{correct} \mid \neg\mathsf{terminate}] && (2.656)\\
=&\Pr[\mathsf{correct} \mid \mathsf{vanish}, \neg\mathsf{terminate}] \cdot \Pr[\mathsf{vanish} \mid \neg\mathsf{terminate}] && (2.657)\\
+&\Pr[\mathsf{correct} \mid \neg\mathsf{vanish}, \neg\mathsf{terminate}] \cdot \Pr[\neg\mathsf{vanish} \mid \neg\mathsf{terminate}] && (2.658)\\
\geq&\Pr[\mathsf{correct} \mid \mathsf{vanish}, \neg\mathsf{terminate}] \cdot \Pr[\mathsf{vanish} \mid \neg\mathsf{terminate}] && (2.659)\\
\overset{\text{Lemma 112}}{\geq}&\Pr[\mathsf{correct} \mid \mathsf{vanish}, \neg\mathsf{terminate}] \cdot \left(1 - \frac{\varepsilon(\lambda)}{\rho(\lambda)}\right) && (2.660)\\
\overset{\text{Lemma 113}}{\geq}&\frac{\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})}{D} \cdot \left(1 - \frac{\varepsilon(\lambda)}{\rho(\lambda)}\right). && (2.661)
\end{aligned}$$

Since $\Pr[\neg\mathsf{terminate}] \geq \rho \cdot Q^{-D-1}$, we can conclude

$$\begin{aligned}
&\Pr[\mathsf{correct}] \tag{2.662}\\
=&\Pr[\mathsf{correct} \mid \mathsf{terminate}] \cdot \Pr[\mathsf{terminate}] \tag{2.663}\\
&+ \Pr[\mathsf{correct} \mid \neg\mathsf{terminate}] \cdot \Pr[\neg\mathsf{terminate}] \tag{2.664}\\
=&\frac{1}{M+1} \cdot \Pr[\mathsf{terminate}] \tag{2.665}\\
&+ \Pr[\mathsf{correct} \mid \neg\mathsf{terminate}] \cdot \Pr[\neg\mathsf{terminate}] \tag{2.666}\\
=&\frac{1}{M+1} \cdot \Pr[\mathsf{terminate}] \tag{2.667}\\
&+ \frac{\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})}{D} \cdot \left(1 - \frac{\varepsilon(\lambda)}{\rho(\lambda)}\right) \cdot \Pr[\neg\mathsf{terminate}] \tag{2.668}\\
\overset{Eq.\ (2.653)}{\geq}&\frac{\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})}{D} \cdot \left(1 - \frac{\varepsilon(\lambda)}{\rho(\lambda)}\right) \cdot \frac{\rho}{Q^{D+1}} + \frac{1}{M+1} \cdot \left(1 - \frac{\rho}{Q^{D+1}}\right) \tag{2.669}\\
=&\left(\frac{\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})}{D} \cdot \left(1 - \frac{\varepsilon(\lambda)}{\rho(\lambda)}\right) - \frac{1}{M+1}\right) \cdot \frac{\rho}{Q^{D+1}} + \frac{1}{M+1}. \tag{2.670}
\end{aligned}$$

Since $x \in \mathcal{X}'_\lambda$ was arbitrary, it follows for the decryption probability of Dec'

$$\mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') \geq \left(\frac{\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})}{D} \cdot \left(1 - \frac{\varepsilon(\lambda)}{\rho(\lambda)}\right) - \frac{1}{M+1}\right) \cdot \frac{\rho}{Q^{D+1}} + \frac{1}{M+1},$$

whenever Eq. (2.653) does hold. For the decryption advantage, we get

$$\begin{aligned}
&\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') \tag{2.671}\\
=&\frac{\#\mathcal{X}'_\lambda \cdot \mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') - 1}{\#\mathcal{X}'_\lambda - 1} \tag{2.672}\\
=&\frac{(M+1) \cdot \mathsf{pr}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') - 1}{(M+1) - 1} \tag{2.673}\\
\geq&\frac{(M+1) \cdot \left(\left(\frac{\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})}{D} \cdot \left(1 - \frac{\varepsilon(\lambda)}{\rho(\lambda)}\right) - \frac{1}{M+1}\right) \cdot \frac{\rho}{Q^{D+1}} + \frac{1}{M+1}\right) - 1}{M} \tag{2.674}\\
=&\frac{(M+1) \cdot \left(\left(\frac{\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})}{D} \cdot \left(1 - \frac{\varepsilon(\lambda)}{\rho(\lambda)}\right) - \frac{1}{M+1}\right) \cdot \frac{\rho}{Q^{D+1}}\right)}{M} \tag{2.675}\\
=&\frac{M+1}{M} \cdot \left(\frac{\mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec})}{D} \cdot \left(1 - \frac{\varepsilon(\lambda)}{\rho(\lambda)}\right) - \frac{1}{M+1}\right) \cdot \frac{\rho}{Q^{D+1}} \qquad \square
\end{aligned}$$

Proof Theorem 108. Assume that FE is correct and secure against the adversary from Algorithm 16 for each $\ell \in \mathsf{poly}(\lambda)$. Then, according to Lemma 114, the decryption advantage of Dec' for SKE' is non-negligible. Choose $a \in \mathsf{poly}(\lambda)$, $a > 0$, s.t. we have for infinitely many $\lambda \in \mathbb{N}$

$$\mathsf{adv}^{\mathsf{dec}}_{\mathsf{SKE}'}(\mathsf{Dec}') \geq \frac{1}{a(\lambda)}. \tag{2.676}$$

Set

$$N' := \left\lceil \lambda \cdot \frac{16 \cdot (d!)^2 \cdot \Gamma_{2d} \cdot (Q-1) \cdot B^2 \cdot (1 + 2 \cdot d! \cdot \Gamma_d \cdot B)^d}{a^{-1} - (d+2) \cdot (Q-1) \cdot (1 - \mathsf{pr}^{\mathsf{dec}}_{\mathsf{FE}}(\mathsf{Dec}))} \right\rceil \in \Theta\big(\lambda a Q B^{2+d}\big).$$

According to Theorem 93 there exists an adversary $\mathcal{A}$ on the selective IND-CPA security of FE that makes $Q-1$ function queries, $3 \cdot {N'}^3$ encryption queries and $\mathsf{poly}(B+\lambda)$ arithmetic operations over $\mathbb{Z}_p$, $\mathbb{Z}_q$ and $\mathbb{Z}$. For the advantage of $\mathcal{A}$, we have

$$\begin{aligned}
&\mathsf{adv}_{\mathsf{FE}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}) && (2.677)\\
\geq &\frac{\mathsf{adv}_{\mathsf{SKE}'}^{\mathsf{dec}}(\mathsf{Dec}') - (d+2)\cdot(Q-1)\cdot(1-\mathsf{pr}_{\mathsf{FE}}^{\mathsf{dec}}(\mathsf{Dec}))}{4d(Q-1)} && (2.678)\\
&- 16\exp(-2N') - (d+4)\cdot {N'}^3 \cdot (Q-1)\cdot(1-\mathsf{pr}_{\mathsf{FE}}^{\mathsf{dec}}(\mathsf{Dec})), && (2.679)
\end{aligned}$$

which is larger than $\frac{1}{4d\cdot a(\lambda)\cdot n(\lambda)^e}$ for λ large enough whenever $\mathsf{adv}_{\mathsf{SKE}'}^{\mathsf{dec}}(\mathsf{Dec}') \geq \frac{1}{a(\lambda)}$. □

Bibliography

[AAB15] Benny Applebaum, Jonathan Avron, and Christina Brzuska. "Arithmetic Cryptography: Extended Abstract". In: *ITCS 2015*. Ed. by Tim Roughgarden. ACM, Jan. 2015, pp. 143–151. DOI: 10.1145/2688073.2688114.

[ABB10] Shweta Agrawal, Dan Boneh, and Xavier Boyen. "Efficient Lattice (H)IBE in the Standard Model". In: *EUROCRYPT 2010*. Ed. by Henri Gilbert. Vol. 6110. LNCS. Springer, Heidelberg, May 2010, pp. 553–572. DOI: 10.1007/978-3-642-13190-5_28.

[ABDP15] Michel Abdalla, Florian Bourse, Angelo De Caro, and David Pointcheval. "Simple Functional Encryption Schemes for Inner Products". In: *PKC 2015*. Ed. by Jonathan Katz. Vol. 9020. LNCS. Springer, Heidelberg, Mar. 2015, pp. 733–751. DOI: 10.1007/978-3-662-46447-2_33.

[ACF09] Michel Abdalla, Dario Catalano, and Dario Fiore. "Verifiable Random Functions from Identity-Based Key Encapsulation". In: *EUROCRYPT 2009*. Ed. by Antoine Joux. Vol. 5479. LNCS. Springer, Heidelberg, Apr. 2009, pp. 554–571. DOI: 10.1007/978-3-642-01001-9_32.

[ACFGU18] Michel Abdalla, Dario Catalano, Dario Fiore, Romain Gay, and Bogdan Ursu. "Multi-Input Functional Encryption for Inner Products: Function-Hiding Realizations and Constructions Without Pairings". In: *CRYPTO 2018, Part I*. Ed. by Hovav Shacham and Alexandra Boldyreva. Vol. 10991. LNCS. Springer, Heidelberg, Aug. 2018, pp. 597–627. DOI: 10.1007/978-3-319-96884-1_20.

[ADINZ17] Benny Applebaum, Ivan Damgård, Yuval Ishai, Michael Nielsen, and Lior Zichron. "Secure Arithmetic Computation with Constant Computational Overhead". In: *CRYPTO 2017, Part I*. Ed. by Jonathan Katz and Hovav Shacham. Vol. 10401. LNCS. Springer, Heidelberg, Aug. 2017, pp. 223–254. DOI: 10.1007/978-3-319-63688-7_8.

[AFHLP16] Martin R. Albrecht, Pooya Farshim, Dennis Hofheinz, Enrique Larraia, and Kenneth G. Paterson. "Multilinear Maps from Obfuscation". In: *TCC 2016-A, Part I*. Ed. by Eyal Kushilevitz and Tal Malkin. Vol. 9562. LNCS. Springer, Heidelberg, Jan. 2016, pp. 446–473. DOI: 10.1007/978-3-662-49096-9_19.

[AFV11] Shweta Agrawal, David Mandell Freeman, and Vinod Vaikuntanathan. "Functional Encryption for Inner Product Predicates from Learning with Errors". In: *ASIACRYPT 2011*. Ed. by Dong Hoon Lee and Xiaoyun Wang. Vol. 7073. LNCS. Springer, Heidelberg, Dec. 2011, pp. 21–40. DOI: 10.1007/978-3-642-25385-0_2.

[AG11] Sanjeev Arora and Rong Ge. "New Algorithms for Learning in Presence of Errors". In: *ICALP 2011, Part I*. Ed. by Luca Aceto, Monika Henzinger, and Jiri Sgall. Vol. 6755. LNCS. Springer, Heidelberg, July 2011, pp. 403–415. DOI: 10.1007/978-3-642-22006-7_34.

[AG90] Eugene L. Allgower and Kurt Georg. *Numerical continuation methods - an introduction*. Vol. 13. Springer series in computational mathematics. Springer, 1990. ISBN: 978-3-540-12760-4. DOI: 10.1007/978-3-642-61257-2.

[AGIS14] Prabhanjan Vijendra Ananth, Divya Gupta, Yuval Ishai, and Amit Sahai. "Optimizing Obfuscation: Avoiding Barrington's Theorem". In: *ACM CCS 2014*. Ed. by Gail-Joon Ahn, Moti Yung, and Ninghui Li. ACM Press, Nov. 2014, pp. 646–658. DOI: 10.1145/2660267.2660342.

[Agr+15] Shashank Agrawal, Shweta Agrawal, Saikrishna Badrinarayanan, Abishek Kumarasubramanian, Manoj Prabhakaran, and Amit Sahai. "On the Practical Security of Inner Product Functional Encryption". In: *PKC 2015*. Ed. by Jonathan Katz. Vol. 9020. LNCS. Springer, Heidelberg, Mar. 2015, pp. 777–798. DOI: 10.1007/978-3-662-46447-2_35.

[Agr19] Shweta Agrawal. "Indistinguishability Obfuscation Without Multilinear Maps: New Methods for Bootstrapping and Instantiation". In: *EUROCRYPT 2019, Part I*. Ed. by Yuval Ishai and Vincent Rijmen. Vol. 11476. LNCS. Springer, Heidelberg, May 2019, pp. 191–225. DOI: 10.1007/978-3-030-17653-2_7.

[AGRW17] Michel Abdalla, Romain Gay, Mariana Raykova, and Hoeteck Wee. "Multi-input Inner-Product Functional Encryption from Pairings". In: *EUROCRYPT 2017, Part I*. Ed. by Jean-Sébastien Coron and Jesper Buus Nielsen. Vol. 10210. LNCS. Springer, Heidelberg, Apr. 2017, pp. 601–626. DOI: 10.1007/978-3-319-56620-7_21.

[AGT21] Shweta Agrawal, Rishab Goyal, and Junichi Tomida. "Multi-Party Functional Encryption". In: *TCC 2021, Part II*. Ed. by Kobbi Nissim and Brent Waters. Vol. 13043. LNCS. Springer, Heidelberg, Nov. 2021, pp. 224–255. DOI: 10.1007/978-3-030-90453-1_8.

[AGVW13] Shweta Agrawal, Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. "Functional Encryption: New Perspectives and Lower Bounds". In: *CRYPTO 2013, Part II*. Ed. by Ran Canetti and Juan A. Garay. Vol. 8043. LNCS. Springer, Heidelberg, Aug. 2013, pp. 500–518. DOI: 10.1007/978-3-642-40084-1_28.

[AH18] Thomas Agrikola and Dennis Hofheinz. “Interactively Secure Groups from Obfuscation”. In: *PKC 2018, Part II*. Ed. by Michel Abdalla and Ricardo Dahab. Vol. 10770. LNCS. Springer, Heidelberg, Mar. 2018, pp. 341–370. DOI: 10.1007/978-3-319-76581-5_12.

[AHK20] Thomas Agrikola, Dennis Hofheinz, and Julia Kastner. “On Instantiating the Algebraic Group Model from Falsifiable Assumptions”. In: *EUROCRYPT 2020, Part II*. Ed. by Anne Canteaut and Yuval Ishai. Vol. 12106. LNCS. Springer, Heidelberg, May 2020, pp. 96–126. DOI: 10.1007/978-3-030-45724-2_4.

[AJ15] Prabhanjan Ananth and Abhishek Jain. “Indistinguishability Obfuscation from Compact Functional Encryption”. In: *CRYPTO 2015, Part I*. Ed. by Rosario Gennaro and Matthew J. B. Robshaw. Vol. 9215. LNCS. Springer, Heidelberg, Aug. 2015, pp. 308–326. DOI: 10.1007/978-3-662-47989-6_15.

[AJLMS19] Prabhanjan Ananth, Aayush Jain, Huijia Lin, Christian Matt, and Amit Sahai. “Indistinguishability Obfuscation Without Multilinear Maps: New Paradigms via Low Degree Weak Pseudorandomness and Security Amplification”. In: *CRYPTO 2019, Part III*. Ed. by Alexandra Boldyreva and Daniele Micciancio. Vol. 11694. LNCS. Springer, Heidelberg, Aug. 2019, pp. 284–332. DOI: 10.1007/978-3-030-26954-8_10.

[Ajt96] Miklós Ajtai. “Generating Hard Instances of Lattice Problems (Extended Abstract)”. In: *28th ACM STOC*. ACM Press, May 1996, pp. 99–108. DOI: 10.1145/237814.237838.

[AK23] Benny Applebaum and Niv Konstantini. “Actively Secure Arithmetic Computation and VOLE with Constant Computational Overhead”. In: *EUROCRYPT 2023, Part II*. Ed. by Carmit Hazay and Martijn Stam. Vol. 14005. LNCS. Springer, Heidelberg, Apr. 2023, pp. 190–219. DOI: 10.1007/978-3-031-30617-4_7.

[AL16] Benny Applebaum and Shachar Lovett. “Algebraic attacks against random local functions and their countermeasures”. In: *48th ACM STOC*. Ed. by Daniel Wichs and Yishay Mansour. ACM Press, June 2016, pp. 1087–1100. DOI: 10.1145/2897518.2897554.

[AL21] Martin R. Albrecht and Russell W. F. Lai. “Subtractive Sets over Cyclotomic Rings - Limits of Schnorr-Like Arguments over Lattices”. In: *CRYPTO 2021, Part II*. Ed. by Tal Malkin and Chris Peikert. Vol. 12826. LNCS. Virtual Event: Springer, Heidelberg, Aug. 2021, pp. 519–548. DOI: 10.1007/978-3-030-84245-1_18.

[Alb+20] Martin R. Albrecht, Pooya Farshim, Shuai Han, Dennis Hofheinz, Enrique Larraia, and Kenneth G. Paterson. “Multilinear Maps from Obfuscation”. In: *Journal of Cryptology* 33.3 (July 2020), pp. 1080–1113. DOI: 10.1007/s00145-019-09340-0.

[Alb10] Martin R. Albrecht. “Algorithmic algebraic techniques and their application to block cipher cryptanalysis”. PhD thesis. Royal Holloway, University of London, Egham, UK, 2010. URL: https://ethos.bl.uk/OrderDetails.do?uin=uk.bl.ethos.529770.

[ALS16] Shweta Agrawal, Benoît Libert, and Damien Stehlé. “Fully Secure Functional Encryption for Inner Products, from Standard Assumptions”. In: *CRYPTO 2016, Part III*. Ed. by Matthew Robshaw and Jonathan Katz. Vol. 9816. LNCS. Springer, Heidelberg, Aug. 2016, pp. 333–362. DOI: 10.1007/978-3-662-53015-3_12.

[AP20] Shweta Agrawal and Alice Pellet-Mary. “Indistinguishability Obfuscation Without Maps: Attacks and Fixes for Noisy Linear FE”. In: *EUROCRYPT 2020, Part I*. Ed. by Anne Canteaut and Yuval Ishai. Vol. 12105. LNCS. Springer, Heidelberg, May 2020, pp. 110–140. DOI: 10.1007/978-3-030-45721-1_5.

[App13] Benny Applebaum. “Cryptographic Hardness of Random Local Functions-Survey”. In: *TCC 2013*. Ed. by Amit Sahai. Vol. 7785. LNCS. Springer, Heidelberg, Mar. 2013, p. 599. DOI: 10.1007/978-3-642-36594-2_33.

[AR17] Shweta Agrawal and Alon Rosen. “Functional Encryption for Bounded Collusions, Revisited”. In: *TCC 2017, Part I*. Ed. by Yael Kalai and Leonid Reyzin. Vol. 10677. LNCS. Springer, Heidelberg, Nov. 2017, pp. 173–205. DOI: 10.1007/978-3-319-70500-2_7.

[AS17] Prabhanjan Ananth and Amit Sahai. “Projective Arithmetic Functional Encryption and Indistinguishability Obfuscation from Degree-5 Multilinear Maps”. In: *EUROCRYPT 2017, Part I*. Ed. by Jean-Sébastien Coron and Jesper Buus Nielsen. Vol. 10210. LNCS. Springer, Heidelberg, Apr. 2017, pp. 152–181. DOI: 10.1007/978-3-319-56620-7_6.

[AV19] Prabhanjan Ananth and Vinod Vaikuntanathan. “Optimal Bounded-Collusion Secure Functional Encryption”. In: *TCC 2019, Part I*. Ed. by Dennis Hofheinz and Alon Rosen. Vol. 11891. LNCS. Springer, Heidelberg, Dec. 2019, pp. 174–198. DOI: 10.1007/978-3-030-36030-6_8.

[Bar+01] Boaz Barak, Oded Goldreich, Russell Impagliazzo, Steven Rudich, Amit Sahai, Salil P. Vadhan, and Ke Yang. “On the (Im)possibility of Obfuscating Programs”. In: *CRYPTO 2001*. Ed. by Joe Kilian. Vol. 2139. LNCS. Springer, Heidelberg, Aug. 2001, pp. 1–18. DOI: 10.1007/3-540-44647-8_1.

[Bar+20] James Bartusek, Yuval Ishai, Aayush Jain, Fermi Ma, Amit Sahai, and Mark Zhandry. “Affine Determinant Programs: A Framework for Obfuscation and Witness Encryption”. In: *ITCS 2020*. Ed. by Thomas Vidick. Vol. 151. LIPIcs, Jan. 2020, 82:1–82:39. DOI: 10.4230/LIPIcs.ITCS.2020.82.

[BBL17] Fabrice Benhamouda, Florian Bourse, and Helger Lipmaa. "CCA-Secure Inner-Product Functional Encryption from Projective Hash Functions". In: *PKC 2017, Part II.* Ed. by Serge Fehr. Vol. 10175. LNCS. Springer, Heidelberg, Mar. 2017, pp. 36–66. DOI: `10.1007/978-3-662-54388-7_2`.

[BCFG17] Carmen Elisabetta Zaira Baltico, Dario Catalano, Dario Fiore, and Romain Gay. "Practical Functional Encryption for Quadratic Functions with Applications to Predicate Encryption". In: *CRYPTO 2017, Part I.* Ed. by Jonathan Katz and Hovav Shacham. Vol. 10401. LNCS. Springer, Heidelberg, Aug. 2017, pp. 67–98. DOI: `10.1007/978-3-319-63688-7_3`.

[BDGM19] Zvika Brakerski, Nico Döttling, Sanjam Garg, and Giulio Malavolta. "Leveraging Linear Decryption: Rate-1 Fully-Homomorphic Encryption and Time-Lock Puzzles". In: *TCC 2019, Part II.* Ed. by Dennis Hofheinz and Alon Rosen. Vol. 11892. LNCS. Springer, Heidelberg, Dec. 2019, pp. 407–437. DOI: `10.1007/978-3-030-36033-7_16`.

[BDGM20] Zvika Brakerski, Nico Döttling, Sanjam Garg, and Giulio Malavolta. "Candidate iO from Homomorphic Encryption Schemes". In: *EUROCRYPT 2020, Part I.* Ed. by Anne Canteaut and Yuval Ishai. Vol. 12105. LNCS. Springer, Heidelberg, May 2020, pp. 79–109. DOI: `10.1007/978-3-030-45721-1_4`.

[BDGM22] Zvika Brakerski, Nico Döttling, Sanjam Garg, and Giulio Malavolta. "Factoring and Pairings Are Not Necessary for IO: Circular-Secure LWE Suffices". In: *ICALP 2022.* Ed. by Mikolaj Bojanczyk, Emanuela Merelli, and David P. Woodruff. Vol. 229. LIPIcs. Schloss Dagstuhl, July 2022, 28:1–28:20. DOI: `10.4230/LIPIcs.ICALP.2022.28`.

[BDWY12] Mihir Bellare, Rafael Dowsley, Brent Waters, and Scott Yilek. "Standard Security Does Not Imply Security against Selective-Opening". In: *EUROCRYPT 2012.* Ed. by David Pointcheval and Thomas Johansson. Vol. 7237. LNCS. Springer, Heidelberg, Apr. 2012, pp. 645–662. DOI: `10.1007/978-3-642-29011-4_38`.

[Ben14] Itaï Ben Yaacov. "A multivariate version of the Vandermonde determinant identity". 2014. URL: `https://hal.science/hal-00983858`.

[Beu21] Ward Beullens. "Improved Cryptanalysis of UOV and Rainbow". In: *EUROCRYPT 2021, Part I.* Ed. by Anne Canteaut and François-Xavier Standaert. Vol. 12696. LNCS. Springer, Heidelberg, Oct. 2021, pp. 348–373. DOI: `10.1007/978-3-030-77870-5_13`.

[Beu22] Ward Beullens. "Breaking Rainbow Takes a Weekend on a Laptop". In: *CRYPTO 2022, Part II.* Ed. by Yevgeniy Dodis and Thomas Shrimpton. Vol. 13508. LNCS. Springer, Heidelberg, Aug. 2022, pp. 464–479. DOI: `10.1007/978-3-031-15979-4_16`.

[BF01] Dan Boneh and Matthew K. Franklin. “Identity-Based Encryption from the Weil Pairing”. In: *CRYPTO 2001*. Ed. by Joe Kilian. Vol. 2139. LNCS. Springer, Heidelberg, Aug. 2001, pp. 213–229. DOI: 10.1007/3-540-44647-8_13.

[BGKPS14] Boaz Barak, Sanjam Garg, Yael Tauman Kalai, Omer Paneth, and Amit Sahai. “Protecting Obfuscation against Algebraic Attacks”. In: *EUROCRYPT 2014*. Ed. by Phong Q. Nguyen and Elisabeth Oswald. Vol. 8441. LNCS. Springer, Heidelberg, May 2014, pp. 221–238. DOI: 10.1007/978-3-642-55220-5_13.

[BHKÜ22] Nicholas Brandt, Dennis Hofheinz, Julia Kastner, and Akin Ünal. “The Price of Verifiability: Lower Bounds for Verifiable Random Functions”. In: *TCC 2022, Part II*. Ed. by Eike Kiltz and Vinod Vaikuntanathan. Vol. 13748. LNCS. Springer, Heidelberg, Nov. 2022, pp. 747–776. DOI: 10.1007/978-3-031-22365-5_26.

[BHY09] Mihir Bellare, Dennis Hofheinz, and Scott Yilek. “Possibility and Impossibility Results for Encryption and Commitment Secure under Selective Opening”. In: *EUROCRYPT 2009*. Ed. by Antoine Joux. Vol. 5479. LNCS. Springer, Heidelberg, Apr. 2009, pp. 1–35. DOI: 10.1007/978-3-642-01001-9_1.

[Bit17] Nir Bitansky. “Verifiable Random Functions from Non-interactive Witness-Indistinguishable Proofs”. In: *TCC 2017, Part II*. Ed. by Yael Kalai and Leonid Reyzin. Vol. 10678. LNCS. Springer, Heidelberg, Nov. 2017, pp. 567–594. DOI: 10.1007/978-3-319-70503-3_19.

[BJK15] Allison Bishop, Abhishek Jain, and Lucas Kowalczyk. “Function-Hiding Inner Product Encryption”. In: *ASIACRYPT 2015, Part I*. Ed. by Tetsu Iwata and Jung Hee Cheon. Vol. 9452. LNCS. Springer, Heidelberg, Nov. 2015, pp. 470–491. DOI: 10.1007/978-3-662-48797-6_20.

[BKW19] Andreas Björklund, Petteri Kaski, and Ryan Williams. “Solving Systems of Polynomial Equations over GF(2) by a Parity-Counting Self-Reduction”. In: *ICALP 2019*. Ed. by Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi. Vol. 132. LIPIcs. Schloss Dagstuhl, July 2019, 26:1–26:13. DOI: 10.4230/LIPIcs.ICALP.2019.26.

[BL17] Xavier Boyen and Qinyi Li. “All-But-Many Lossy Trapdoor Functions from Lattices and Applications”. In: *CRYPTO 2017, Part III*. Ed. by Jonathan Katz and Hovav Shacham. Vol. 10403. LNCS. Springer, Heidelberg, Aug. 2017, pp. 298–331. DOI: 10.1007/978-3-319-63697-9_11.

[BM23] Zvika Brakerski and Stav Medina. *Limits on Adaptive Security for Attribute-Based Encryption*. Cryptology ePrint Archive, Paper 2023/952. June 2023. URL: https://eprint.iacr.org/2023/952.

[BMR10] Dan Boneh, Hart William Montgomery, and Ananth Raghunathan. "Algebraic pseudorandom functions with improved efficiency from the augmented cascade". In: *ACM CCS 2010*. Ed. by Ehab Al-Shaer, Angelos D. Keromytis, and Vitaly Shmatikov. ACM Press, Oct. 2010, pp. 131–140. DOI: `10.1145/1866307.1866323`.

[BMSV22] Emanuele Bellini, Rusydi H. Makarim, Carlo Sanna, and Javier A. Verbel. "An Estimator for the Hardness of the MQ Problem". In: *AFRICACRYPT 22*. Ed. by Lejla Batina and Joan Daemen. Vol. 2022. LNCS. Springer Nature, July 2022, pp. 323–347. DOI: `10.1007/978-3-031-17433-9_14`.

[BMSZ16] Saikrishna Badrinarayanan, Eric Miles, Amit Sahai, and Mark Zhandry. "Post-zeroizing Obfuscation: New Mathematical Tools, and the Case of Evasive Circuits". In: *EUROCRYPT 2016, Part II*. Ed. by Marc Fischlin and Jean-Sébastien Coron. Vol. 9666. LNCS. Springer, Heidelberg, May 2016, pp. 764–791. DOI: `10.1007/978-3-662-49896-5_27`.

[Bon+14] Dan Boneh, Craig Gentry, Sergey Gorbunov, Shai Halevi, Valeria Nikolaenko, Gil Segev, Vinod Vaikuntanathan, and Dhinakaran Vinayagamurthy. "Fully Key-Homomorphic Encryption, Arithmetic Circuit ABE and Compact Garbled Circuits". In: *EUROCRYPT 2014*. Ed. by Phong Q. Nguyen and Elisabeth Oswald. Vol. 8441. LNCS. Springer, Heidelberg, May 2014, pp. 533–556. DOI: `10.1007/978-3-642-55220-5_30`.

[BR14] Zvika Brakerski and Guy N. Rothblum. "Virtual Black-Box Obfuscation for All Circuits via Generic Graded Encoding". In: *TCC 2014*. Ed. by Yehuda Lindell. Vol. 8349. LNCS. Springer, Heidelberg, Feb. 2014, pp. 1–25. DOI: `10.1007/978-3-642-54242-8_1`.

[BR93] Mihir Bellare and Phillip Rogaway. "Random Oracles are Practical: A Paradigm for Designing Efficient Protocols". In: *ACM CCS 93*. Ed. by Dorothy E. Denning, Raymond Pyle, Ravi Ganesan, Ravi S. Sandhu, and Victoria Ashby. ACM Press, Nov. 1993, pp. 62–73. DOI: `10.1145/168588.168596`.

[Bra+15] Zvika Brakerski, Craig Gentry, Shai Halevi, Tancrède Lepoint, Amit Sahai, and Mehdi Tibouchi. *Cryptanalysis of the Quadratic Zero-Testing of GGH*. Cryptology ePrint Archive, Report 2015/845. `https://eprint.iacr.org/2015/845`. 2015.

[BRS13a] Dan Boneh, Ananth Raghunathan, and Gil Segev. "Function-Private Identity-Based Encryption: Hiding the Function in Functional Encryption". In: *CRYPTO 2013, Part II*. Ed. by Ran Canetti and Juan A. Garay. Vol. 8043. LNCS. Springer, Heidelberg, Aug. 2013, pp. 461–478. DOI: `10.1007/978-3-642-40084-1_26`.

[BRS13b] Dan Boneh, Ananth Raghunathan, and Gil Segev. "Function-Private Subspace-Membership Encryption and Its Applications". In: *ASIACRYPT 2013, Part I.* Ed. by Kazue Sako and Palash Sarkar. Vol. 8269. LNCS. Springer, Heidelberg, Dec. 2013, pp. 255–275. DOI: 10.1007/978-3-642-42033-7_14.

[BS15] Zvika Brakerski and Gil Segev. "Function-Private Functional Encryption in the Private-Key Setting". In: *TCC 2015, Part II.* Ed. by Yevgeniy Dodis and Jesper Buus Nielsen. Vol. 9015. LNCS. Springer, Heidelberg, Mar. 2015, pp. 306–324. DOI: 10.1007/978-3-662-46497-7_12.

[BSW11] Dan Boneh, Amit Sahai, and Brent Waters. "Functional Encryption: Definitions and Challenges". In: *TCC 2011.* Ed. by Yuval Ishai. Vol. 6597. LNCS. Springer, Heidelberg, Mar. 2011, pp. 253–273. DOI: 10.1007/978-3-642-19571-6_16.

[Buc76] B. Buchberger. "A Theoretical Basis for the Reduction of Polynomials to Canonical Forms". In: *SIGSAM Bull.* 10.3 (Aug. 1976), pp. 19–29. ISSN: 0163-5824. DOI: 10.1145/1088216.1088219.

[BV11] Zvika Brakerski and Vinod Vaikuntanathan. "Efficient Fully Homomorphic Encryption from (Standard) LWE". In: *52nd FOCS.* Ed. by Rafail Ostrovsky. IEEE Computer Society Press, Oct. 2011, pp. 97–106. DOI: 10.1109/FOCS.2011.12.

[BV15] Nir Bitansky and Vinod Vaikuntanathan. "Indistinguishability Obfuscation from Functional Encryption". In: *56th FOCS.* Ed. by Venkatesan Guruswami. IEEE Computer Society Press, Oct. 2015, pp. 171–190. DOI: 10.1109/FOCS.2015.20.

[BW07] Dan Boneh and Brent Waters. "Conjunctive, Subset, and Range Queries on Encrypted Data". In: *TCC 2007.* Ed. by Salil P. Vadhan. Vol. 4392. LNCS. Springer, Heidelberg, Feb. 2007, pp. 535–554. DOI: 10.1007/978-3-540-70936-7_29.

[BWZ14] Dan Boneh, David J. Wu, and Joe Zimmerman. *Immunizing Multilinear Maps Against Zeroizing Attacks.* Cryptology ePrint Archive, Report 2014/930. https://eprint.iacr.org/2014/930. 2014.

[CCNY12] Chen-Mou Cheng, Tung Chou, Ruben Niederhagen, and Bo-Yin Yang. "Solving Quadratic Equations with XL on Parallel Architectures". In: *CHES 2012.* Ed. by Emmanuel Prouff and Patrick Schaumont. Vol. 7428. LNCS. Springer, Heidelberg, Sept. 2012, pp. 356–373. DOI: 10.1007/978-3-642-33027-8_21.

[CDGPP18] Jérémy Chotard, Edouard Dufour Sans, Romain Gay, Duong Hieu Phan, and David Pointcheval. "Decentralized Multi-Client Functional Encryption for Inner Product". In: *ASIACRYPT 2018, Part II.* Ed. by Thomas Peyrin and Steven Galbraith. Vol. 11273. LNCS. Springer, Heidelberg, Dec. 2018, pp. 703–732. DOI: 10.1007/978-3-030-03329-3_24.

[CFLMR16] Jung Hee Cheon, Pierre-Alain Fouque, Changmin Lee, Brice Minaud, and Hansol Ryu. “Cryptanalysis of the New CLT Multilinear Map over the Integers”. In: *EUROCRYPT 2016, Part I.* Ed. by Marc Fischlin and Jean-Sébastien Coron. Vol. 9665. LNCS. Springer, Heidelberg, May 2016, pp. 509–536. DOI: 10.1007/978-3-662-49890-3_20.

[CG21] Alessio Caminata and Elisa Gorla. “Solving Multivariate Polynomial Systems and an Invariant from Commutative Algebra”. In: *Arithmetic of Finite Fields.* Ed. by Jean Claude Bajard and Alev Topuzoğlu. Cham: Springer International Publishing, 2021, pp. 3–36. ISBN: 978-3-030-68869-1. DOI: 10.1007/978-3-030-68869-1_1.

[CG23] Alessio Caminata and Elisa Gorla. “Solving Degree, Last Fall Degree, and Related Invariants”. In: *J. Symb. Comput.* 114.C (2023), pp. 322–335. ISSN: 0747-7171. DOI: 10.1016/j.jsc.2022.05.001.

[CHKP10] David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert. “Bonsai Trees, or How to Delegate a Lattice Basis”. In: *EUROCRYPT 2010.* Ed. by Henri Gilbert. Vol. 6110. LNCS. Springer, Heidelberg, May 2010, pp. 523–552. DOI: 10.1007/978-3-642-13190-5_27.

[CHLRS15] Jung Hee Cheon, Kyoohyung Han, Changmin Lee, Hansol Ryu, and Damien Stehlé. “Cryptanalysis of the Multilinear Map over the Integers”. In: *EUROCRYPT 2015, Part I.* Ed. by Elisabeth Oswald and Marc Fischlin. Vol. 9056. LNCS. Springer, Heidelberg, Apr. 2015, pp. 3–12. DOI: 10.1007/978-3-662-46800-5_1.

[CKPS00] Nicolas Courtois, Alexander Klimov, Jacques Patarin, and Adi Shamir. “Efficient Algorithms for Solving Overdefined Systems of Multivariate Polynomial Equations”. In: *EUROCRYPT 2000.* Ed. by Bart Preneel. Vol. 1807. LNCS. Springer, Heidelberg, May 2000, pp. 392–407. DOI: 10.1007/3-540-45539-6_27.

[CLT13] Jean-Sébastien Coron, Tancrède Lepoint, and Mehdi Tibouchi. “Practical Multilinear Maps over the Integers”. In: *CRYPTO 2013, Part I.* Ed. by Ran Canetti and Juan A. Garay. Vol. 8042. LNCS. Springer, Heidelberg, Aug. 2013, pp. 476–493. DOI: 10.1007/978-3-642-40041-4_26.

[CLT15] Jean-Sébastien Coron, Tancrède Lepoint, and Mehdi Tibouchi. “New Multilinear Maps Over the Integers”. In: *CRYPTO 2015, Part I.* Ed. by Rosario Gennaro and Matthew J. B. Robshaw. Vol. 9215. LNCS. Springer, Heidelberg, Aug. 2015, pp. 267–286. DOI: 10.1007/978-3-662-47989-6_13.

[Coc01] Clifford Cocks. “An Identity Based Encryption Scheme Based on Quadratic Residues”. In: *8th IMA International Conference on Cryptography and Coding.* Ed. by Bahram Honary. Vol. 2260. LNCS. Springer, Heidelberg, Dec. 2001, pp. 360–363.

[Cop94] Don Coppersmith. “Solving Homogeneous Linear Equations Over GF(2) via Block Wiedemann Algorithm”. In: *Mathematics of Computation* 62.205 (1994), pp. 333–350. ISSN: 00255718, 10886842. DOI: 10.2307/2153413.

[Cor02] Jean-Sébastien Coron. “Optimal Security Proofs for PSS and Other Signature Schemes”. In: *EUROCRYPT 2002*. Ed. by Lars R. Knudsen. Vol. 2332. LNCS. Springer, Heidelberg, Apr. 2002, pp. 272–287. DOI: 10.1007/3-540-46035-7_18.

[DBMMW08] Jintai Ding, Johannes Buchmann, Mohamed Saied Emam Mohamed, Wael Said Abd Elmageed Mohamed, and Ralf-Philipp Weinmann. “MutantXL”. de. In: *SCC* TUD-CS-2009-0142 (Jan. 2008), pp. 16–22. URL: http://tubiblio.ulb.tu-darmstadt.de/100617/.

[DDM16] Pratish Datta, Ratna Dutta, and Sourav Mukhopadhyay. “Functional Encryption for Inner Product with Full Function Privacy”. In: *PKC 2016, Part I*. Ed. by Chen-Mou Cheng, Kai-Min Chung, Giuseppe Persiano, and Bo-Yin Yang. Vol. 9614. LNCS. Springer, Heidelberg, Mar. 2016, pp. 164–195. DOI: 10.1007/978-3-662-49384-7_7.

[De +13] Angelo De Caro, Vincenzo Iovino, Abhishek Jain, Adam O’Neill, Omer Paneth, and Giuseppe Persiano. “On the Achievability of Simulation-Based Security for Functional Encryption”. In: *CRYPTO 2013, Part II*. Ed. by Ran Canetti and Juan A. Garay. Vol. 8043. LNCS. Springer, Heidelberg, Aug. 2013, pp. 519–535. DOI: 10.1007/978-3-642-40084-1_29.

[DGGMM18] Nico Döttling, Sanjam Garg, Divya Gupta, Peihan Miao, and Pratyay Mukherjee. “Obfuscation from Low Noise Multilinear Maps”. In: *INDOCRYPT 2018*. Ed. by Debrup Chakraborty and Tetsu Iwata. Vol. 11356. LNCS. Springer, Heidelberg, Dec. 2018, pp. 329–352. DOI: 10.1007/978-3-030-05378-9_18.

[DGW07] Zeev Dvir, Ariel Gabizon, and Avi Wigderson. “Extractors and Rank Extractors for Polynomial Sources”. In: *48th FOCS*. IEEE Computer Society Press, Oct. 2007, pp. 52–62. DOI: 10.1109/FOCS.2007.26.

[DH76] W. Diffie and M. Hellman. “New directions in cryptography”. In: *IEEE Transactions on Information Theory* 22.6 (1976), pp. 644–654. DOI: 10.1109/TIT.1976.1055638.

[Die04] Claus Diem. “The XL-Algorithm and a Conjecture from Commutative Algebra”. In: *ASIACRYPT 2004*. Ed. by Pil Joong Lee. Vol. 3329. LNCS. Springer, Heidelberg, Dec. 2004, pp. 323–337. DOI: 10.1007/978-3-540-30539-2_23.

[Din21a] Itai Dinur. “Cryptanalytic Applications of the Polynomial Method for Solving Multivariate Equation Systems over GF(2)”. In: *EUROCRYPT 2021, Part I*. Ed. by Anne Canteaut and François-Xavier Standaert. Vol. 12696. LNCS. Springer, Heidelberg, Oct. 2021, pp. 374–403. DOI: 10.1007/978-3-030-77870-5_14.

[Din21b] Itai Dinur. “Improved Algorithms for Solving Polynomial Systems over GF(2) by Multiple Parity-Counting”. In: *32nd SODA*. Ed. by Dániel Marx. ACM-SIAM, Jan. 2021, pp. 2550–2564. DOI: 10.1137/1.9781611976465.151.

[DL78] Richard A. Demillo and Richard J. Lipton. “A probabilistic remark on algebraic program testing”. In: *Information Processing Letters* 7.4 (1978), pp. 193–195. ISSN: 0020-0190. DOI: 10.1016/0020-0190(78)90067-4.

[DNRS99] Cynthia Dwork, Moni Naor, Omer Reingold, and Larry J. Stockmeyer. “Magic Functions”. In: *40th FOCS*. IEEE Computer Society Press, Oct. 1999, pp. 523–534. DOI: 10.1109/SFFCS.1999.814626.

[Dod03] Yevgeniy Dodis. “Efficient Construction of (Distributed) Verifiable Random Functions”. In: *PKC 2003*. Ed. by Yvo Desmedt. Vol. 2567. LNCS. Springer, Heidelberg, Jan. 2003, pp. 1–17. DOI: 10.1007/3-540-36288-6_1.

[DU14] Stefano De Marchi and Konstantin Usevich. “On certain multivariate Vandermonde determinants whose variables separate”. In: *Linear Algebra and its Applications* 449 (2014), pp. 17–27. ISSN: 0024-3795. DOI: 10.1016/j.laa.2014.01.034.

[Dub90] Thomas W. Dubé. “The Structure of Polynomial Ideals and Gröbner Bases”. In: *SIAM Journal on Computing* 19.4 (1990), pp. 750–773. DOI: 10.1137/0219053.

[DY05] Yevgeniy Dodis and Aleksandr Yampolskiy. “A Verifiable Random Function with Short Proofs and Keys”. In: *PKC 2005*. Ed. by Serge Vaudenay. Vol. 3386. LNCS. Springer, Heidelberg, Jan. 2005, pp. 416–431. DOI: 10.1007/978-3-540-30580-4_28.

[Fau02] Jean Charles Faugère. “A New Efficient Algorithm for Computing Gröbner Bases without Reduction to Zero (F5)”. In: *Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation*. ISSAC ’02. Lille, France: Association for Computing Machinery, 2002, pp. 75–83. ISBN: 1581134843. DOI: 10.1145/780506.780516.

[Fau99] Jean-Charles Faugére. “A new efficient algorithm for computing Gröbner bases (F4)”. In: *Journal of Pure and Applied Algebra* 139.1 (1999), pp. 61–88. ISSN: 0022-4049. DOI: 10.1016/S0022-4049(99)00005-5.

[FHHL18] Pooya Farshim, Julia Hesse, Dennis Hofheinz, and Enrique Larraia. “Graded Encoding Schemes from Obfuscation”. In: *PKC 2018, Part II*. Ed. by Michel Abdalla and Ricardo Dahab. Vol. 10770. LNCS. Springer, Heidelberg, Mar. 2018, pp. 371–400. DOI: 10.1007/978-3-319-76581-5_13.

[FKL18] Georg Fuchsbauer, Eike Kiltz, and Julian Loss. “The Algebraic Group Model and its Applications”. In: *CRYPTO 2018, Part II*. Ed. by Hovav Shacham and Alexandra Boldyreva. Vol. 10992. LNCS. Springer, Heidelberg, Aug. 2018, pp. 33–62. DOI: 10.1007/978-3-319-96881-0_2.

[FMR99] G. Frey, M. Muller, and H.-G. Ruck. “The Tate pairing and the discrete logarithm applied to elliptic curve cryptosystems”. In: *IEEE Transactions on Information Theory* 45.5 (1999), pp. 1717–1719. DOI: 10.1109/18.771254.

[FR94] Gerhard Frey and Hans-Georg Rück. “A Remark Concerning m-Divisibility and the Discrete Logarithm in the Divisor Class Group of Curves”. In: *Mathematics of Computation* 62.206 (1994), pp. 865–874. ISSN: 00255718, 10886842. DOI: 10.2307/2153546.

[Gar+13] Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. “Candidate Indistinguishability Obfuscation and Functional Encryption for all Circuits”. In: *54th FOCS*. IEEE Computer Society Press, Oct. 2013, pp. 40–49. DOI: 10.1109/FOCS.2013.13.

[Gay19] Romain Gay. “Public-key encryption, revisited : tight security and richer functionalities”. Theses. Université Paris sciences et lettres, Mar. 2019. URL: https://theses.hal.science/tel-03416070.

[Gay20] Romain Gay. “A New Paradigm for Public-Key Functional Encryption for Degree-2 Polynomials”. In: *PKC 2020, Part I*. Ed. by Aggelos Kiayias, Markulf Kohlweiss, Petros Wallden, and Vassilis Zikas. Vol. 12110. LNCS. Springer, Heidelberg, May 2020, pp. 95–120. DOI: 10.1007/978-3-030-45374-9_4.

[Gen09] Craig Gentry. “Fully homomorphic encryption using ideal lattices”. In: *41st ACM STOC*. Ed. by Michael Mitzenmacher. ACM Press, May 2009, pp. 169–178. DOI: 10.1145/1536414.1536440.

[GGH13] Sanjam Garg, Craig Gentry, and Shai Halevi. “Candidate Multilinear Maps from Ideal Lattices”. In: *EUROCRYPT 2013*. Ed. by Thomas Johansson and Phong Q. Nguyen. Vol. 7881. LNCS. Springer, Heidelberg, May 2013, pp. 1–17. DOI: 10.1007/978-3-642-38348-9_1.

[GGH15] Craig Gentry, Sergey Gorbunov, and Shai Halevi. “Graph-Induced Multilinear Maps from Lattices”. In: *TCC 2015, Part II*. Ed. by Yevgeniy Dodis and Jesper Buus Nielsen. Vol. 9015. LNCS. Springer, Heidelberg, Mar. 2015, pp. 498–527. DOI: 10.1007/978-3-662-46497-7_20.

[GGHSW13] Sanjam Garg, Craig Gentry, Shai Halevi, Amit Sahai, and Brent Waters. “Attribute-Based Encryption for Circuits from Multilinear Maps”. In: *CRYPTO 2013, Part II*. Ed. by Ran Canetti and Juan A. Garay. Vol. 8043. LNCS. Springer, Heidelberg, Aug. 2013, pp. 479–499. DOI: 10.1007/978-3-642-40084-1_27.

[GHKW17] Rishab Goyal, Susan Hohenberger, Venkata Koppula, and Brent Waters. “A Generic Approach to Constructing and Proving Verifiable Random Functions”. In: *TCC 2017, Part II*. Ed. by Yael Kalai and Leonid Reyzin. Vol. 10678. LNCS. Springer, Heidelberg, Nov. 2017, pp. 537–566. DOI: 10.1007/978-3-319-70503-3_18.

[GJK18] Craig Gentry, Charanjit S. Jutla, and Daniel Kane. *Obfuscation Using Tensor Products*. Cryptology ePrint Archive, Report 2018/756. https://eprint.iacr.org/2018/756. 2018.

[GJLS21] Romain Gay, Aayush Jain, Huijia Lin, and Amit Sahai. "Indistinguishability Obfuscation from Simple-to-State Hard Problems: New Assumptions, New Techniques, and Simplification". In: *EUROCRYPT 2021, Part III*. Ed. by Anne Canteaut and François-Xavier Standaert. Vol. 12698. LNCS. Springer, Heidelberg, Oct. 2021, pp. 97–126. DOI: 10.1007/978-3-030-77883-5_4.

[GKPVZ13] Shafi Goldwasser, Yael Tauman Kalai, Raluca A. Popa, Vinod Vaikuntanathan, and Nickolai Zeldovich. "Reusable garbled circuits and succinct functional encryption". In: *45th ACM STOC*. Ed. by Dan Boneh, Tim Roughgarden, and Joan Feigenbaum. ACM Press, June 2013, pp. 555–564. DOI: 10.1145/2488608.2488678.

[GKRS20] Siyao Guo, Pritish Kamath, Alon Rosen, and Katerina Sotiraki. "Limits on the Efficiency of (Ring) LWE Based Non-interactive Key Exchange". In: *PKC 2020, Part I*. Ed. by Aggelos Kiayias, Markulf Kohlweiss, Petros Wallden, and Vassilis Zikas. Vol. 12110. LNCS. Springer, Heidelberg, May 2020, pp. 374–395. DOI: 10.1007/978-3-030-45374-9_13.

[Gol+14] Shafi Goldwasser, S. Dov Gordon, Vipul Goyal, Abhishek Jain, Jonathan Katz, Feng-Hao Liu, Amit Sahai, Elaine Shi, and Hong-Sheng Zhou. "Multi-input Functional Encryption". In: *EUROCRYPT 2014*. Ed. by Phong Q. Nguyen and Elisabeth Oswald. Vol. 8441. LNCS. Springer, Heidelberg, May 2014, pp. 578–602. DOI: 10.1007/978-3-642-55220-5_32.

[Gol11] Oded Goldreich. "Candidate One-Way Functions Based on Expander Graphs". In: *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation: In Collaboration with Lidor Avigad, Mihir Bellare, Zvika Brakerski, Shafi Goldwasser, Shai Halevi, Tali Kaufman, Leonid Levin, Noam Nisan, Dana Ron, Madhu Sudan, Luca Trevisan, Salil Vadhan, Avi Wigderson, David Zuckerman*. Ed. by Oded Goldreich. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 76–87. ISBN: 978-3-642-22670-0. DOI: 10.1007/978-3-642-22670-0_10.

[GP21] Romain Gay and Rafael Pass. "Indistinguishability obfuscation from circular security". In: *53rd ACM STOC*. Ed. by Samir Khuller and Virginia Vassilevska Williams. ACM Press, June 2021, pp. 736–749. DOI: 10.1145/3406325.3451070.

[GPS08] Steven D. Galbraith, Kenneth G. Paterson, and Nigel P. Smart. "Pairings for cryptographers". In: *Discrete Applied Mathematics* 156.16 (2008). Applications of Algebra to Cryptography, pp. 3113–3121. ISSN: 0166-218X. DOI: 10.1016/j.dam.2007.12.010.

[GPSW06] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data". In: *ACM CCS 2006*. Ed. by Ari Juels, Rebecca N. Wright, and Sabrina De Capitani di Vimercati. Available as Cryptology ePrint Archive Report 2006/309. ACM Press, Oct. 2006, pp. 89–98. DOI: `10.1145/1180405.1180418`.

[GS02] Craig Gentry and Alice Silverberg. "Hierarchical ID-Based Cryptography". In: *ASIACRYPT 2002*. Ed. by Yuliang Zheng. Vol. 2501. LNCS. Springer, Heidelberg, Dec. 2002, pp. 548–566. DOI: `10.1007/3-540-36178-2_34`.

[GSW13] Craig Gentry, Amit Sahai, and Brent Waters. "Homomorphic Encryption from Learning with Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based". In: *CRYPTO 2013, Part I*. Ed. by Ran Canetti and Juan A. Garay. Vol. 8042. LNCS. Springer, Heidelberg, Aug. 2013, pp. 75–92. DOI: `10.1007/978-3-642-40041-4_5`.

[GVW12] Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. "Functional Encryption with Bounded Collusions via Multi-party Computation". In: *CRYPTO 2012*. Ed. by Reihaneh Safavi-Naini and Ran Canetti. Vol. 7417. LNCS. Springer, Heidelberg, Aug. 2012, pp. 162–179. DOI: `10.1007/978-3-642-32009-5_11`.

[GVW13] Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. "Attribute-based encryption for circuits". In: *45th ACM STOC*. Ed. by Dan Boneh, Tim Roughgarden, and Joan Feigenbaum. ACM Press, June 2013, pp. 545–554. DOI: `10.1145/2488608.2488677`.

[GVW15] Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. "Predicate Encryption for Circuits from LWE". In: *CRYPTO 2015, Part II*. Ed. by Rosario Gennaro and Matthew J. B. Robshaw. Vol. 9216. LNCS. Springer, Heidelberg, Aug. 2015, pp. 503–523. DOI: `10.1007/978-3-662-48000-7_25`.

[HHKKÜ23] Dennis Hofheinz, Kristina Hostáková, Julia Kastner, Karen Klein, and Akin Ünal. *Compact Lossy Trapdoor Functions and Selective Opening Security From LWE*. Cryptology ePrint Archive, Paper 2023/864. June 2023. URL: `https://eprint.iacr.org/2023/864`.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. "A Pseudorandom Generator from any One-way Function". In: *SIAM Journal on Computing* 28.4 (1999), pp. 1364–1396.

[HJ16] Dennis Hofheinz and Tibor Jager. "Verifiable Random Functions from Standard Assumptions". In: *TCC 2016-A, Part I*. Ed. by Eyal Kushilevitz and Tal Malkin. Vol. 9562. LNCS. Springer, Heidelberg, Jan. 2016, pp. 336–362. DOI: `10.1007/978-3-662-49096-9_14`.

[HJS16] Gavin Harrison, Jeremy Johnson, and B. David Saunders. "Probabilistic analysis of Wiedemann's algorithm for minimal polynomial computation". In: *Journal of Symbolic Computation* 74 (2016), pp. 55–69. ISSN: 0747-7171. DOI: 10.1016/j.jsc.2015.06.005.

[HJS22] Gavin Harrison, Jeremy Johnson, and B. David Saunders. "Probabilistic analysis of block Wiedemann for leading invariant factors". In: *Journal of Symbolic Computation* 108 (2022), pp. 98–116. ISSN: 0747-7171. DOI: 10.1016/j.jsc.2021.06.005.

[HL02] Jeremy Horwitz and Ben Lynn. "Toward Hierarchical Identity-Based Encryption". In: *EUROCRYPT 2002*. Ed. by Lars R. Knudsen. Vol. 2332. LNCS. Springer, Heidelberg, Apr. 2002, pp. 466–481. DOI: 10.1007/3-540-46035-7_31.

[HMS07] Dennis Hofheinz, John Malone-Lee, and Martijn Stam. "Obfuscation for Cryptographic Purposes". In: *TCC 2007*. Ed. by Salil P. Vadhan. Vol. 4392. LNCS. Springer, Heidelberg, Feb. 2007, pp. 214–232. DOI: 10.1007/978-3-540-70936-7_12.

[HMS10] Dennis Hofheinz, John Malone-Lee, and Martijn Stam. "Obfuscation for Cryptographic Purposes". In: *Journal of Cryptology* 23.1 (Jan. 2010), pp. 121–168. DOI: 10.1007/s00145-009-9046-1.

[HO13] Brett Hemenway and Rafail Ostrovsky. "Building Lossy Trapdoor Functions from Lossy Encryption". In: *ASIACRYPT 2013, Part II*. Ed. by Kazue Sako and Palash Sarkar. Vol. 8270. LNCS. Springer, Heidelberg, Dec. 2013, pp. 241–260. DOI: 10.1007/978-3-642-42045-0_13.

[Hoe63] Wassily Hoeffding. "Probability Inequalities for Sums of Bounded Random Variables". In: *Journal of the American Statistical Association* 58.301 (1963), pp. 13–30. ISSN: 01621459. DOI: 10.2307/2282952.

[Hof12] Dennis Hofheinz. "All-But-Many Lossy Trapdoor Functions". In: *EUROCRYPT 2012*. Ed. by David Pointcheval and Thomas Johansson. Vol. 7237. LNCS. Springer, Heidelberg, Apr. 2012, pp. 209–227. DOI: 10.1007/978-3-642-29011-4_14.

[HR14] Dennis Hofheinz and Andy Rupp. "Standard versus Selective Opening Security: Separation and Equivalence Results". In: *TCC 2014*. Ed. by Yehuda Lindell. Vol. 8349. LNCS. Springer, Heidelberg, Feb. 2014, pp. 591–615. DOI: 10.1007/978-3-642-54242-8_25.

[HRW16] Dennis Hofheinz, Vanishree Rao, and Daniel Wichs. "Standard Security Does Not Imply Indistinguishability Under Selective Opening". In: *TCC 2016-B, Part II*. Ed. by Martin Hirt and Adam D. Smith. Vol. 9986. LNCS. Springer, Heidelberg, Oct. 2016, pp. 121–145. DOI: 10.1007/978-3-662-53644-5_5.

[HU19] Dennis Hofheinz and Bogdan Ursu. “Dual-Mode NIZKs from Obfuscation”. In: *ASIACRYPT 2019, Part I*. Ed. by Steven D. Galbraith and Shiho Moriai. Vol. 11921. LNCS. Springer, Heidelberg, Dec. 2019, pp. 311–341. DOI: 10.1007/978-3-030-34578-5_12.

[Huy86] Dung T. Huynh. “A superexponential lower bound for Gröbner bases and Church-Rosser commutative thue systems”. In: *Information and Control* 68.1 (1986), pp. 196–206. ISSN: 0019-9958. DOI: 10.1016/S0019-9958(86)80035-3.

[HW10] Susan Hohenberger and Brent Waters. “Constructing Verifiable Random Functions with Large Input Spaces”. In: *EUROCRYPT 2010*. Ed. by Henri Gilbert. Vol. 6110. LNCS. Springer, Heidelberg, May 2010, pp. 656–672. DOI: 10.1007/978-3-642-13190-5_33.

[HW14] Susan Hohenberger and Brent Waters. “Online/Offline Attribute-Based Encryption”. In: *PKC 2014*. Ed. by Hugo Krawczyk. Vol. 8383. LNCS. Springer, Heidelberg, Mar. 2014, pp. 293–310. DOI: 10.1007/978-3-642-54631-0_17.

[IKOS08] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. “Cryptography with constant computational overhead”. In: *40th ACM STOC*. Ed. by Richard E. Ladner and Cynthia Dwork. ACM Press, May 2008, pp. 433–442. DOI: 10.1145/1374376.1374438.

[IPS08] Yuval Ishai, Manoj Prabhakaran, and Amit Sahai. “Founding Cryptography on Oblivious Transfer - Efficiently”. In: *CRYPTO 2008*. Ed. by David Wagner. Vol. 5157. LNCS. Springer, Heidelberg, Aug. 2008, pp. 572–591. DOI: 10.1007/978-3-540-85174-5_32.

[IW97] Russell Impagliazzo and Avi Wigderson. “P = BPP if E Requires Exponential Circuits: Derandomizing the XOR Lemma”. In: *29th ACM STOC*. ACM Press, May 1997, pp. 220–229. DOI: 10.1145/258533.258590.

[Jag15] Tibor Jager. “Verifiable Random Functions from Weaker Assumptions”. In: *TCC 2015, Part II*. Ed. by Yevgeniy Dodis and Jesper Buus Nielsen. Vol. 9015. LNCS. Springer, Heidelberg, Mar. 2015, pp. 121–143. DOI: 10.1007/978-3-662-46497-7_5.

[JLMS19] Aayush Jain, Huijia Lin, Christian Matt, and Amit Sahai. “How to Leverage Hardness of Constant-Degree Expanding Polynomials over $\mathbb{R}$ to build $i\mathcal{O}$”. In: *EUROCRYPT 2019, Part I*. Ed. by Yuval Ishai and Vincent Rijmen. Vol. 11476. LNCS. Springer, Heidelberg, May 2019, pp. 251–281. DOI: 10.1007/978-3-030-17653-2_9.

[JLS21] Aayush Jain, Huijia Lin, and Amit Sahai. “Indistinguishability obfuscation from well-founded assumptions”. In: *53rd ACM STOC*. Ed. by Samir Khuller and Virginia Vassilevska Williams. ACM Press, June 2021, pp. 60–73. DOI: 10.1145/3406325.3451093.

[JLS22] Aayush Jain, Huijia Lin, and Amit Sahai. "Indistinguishability Obfuscation from LPN over $\mathbb{F}_p$, DLIN, and PRGs in NC^0". In: *EUROCRYPT 2022, Part I*. Ed. by Orr Dunkelman and Stefan Dziembowski. Vol. 13275. LNCS. Springer, Heidelberg, May 2022, pp. 670–699. DOI: 10.1007/978-3-031-06944-4_23.

[JN03] Antoine Joux and Kim Nguyen. "Separating Decision Diffie-Hellman from Computational Diffie-Hellman in Cryptographic Groups". In: *Journal of Cryptology* 16.4 (Sept. 2003), pp. 239–247. DOI: 10.1007/s00145-003-0052-4.

[Jou04] Antoine Joux. "A One Round Protocol for Tripartite Diffie-Hellman". In: *Journal of Cryptology* 17.4 (Sept. 2004), pp. 263–276. DOI: 10.1007/s00145-004-0312-y.

[Kal95] Erich Kaltofen. "Analysis of Coppersmith's Block Wiedemann Algorithm for the Parallel Solution of Sparse Linear Systems". In: *Mathematics of Computation* 64.210 (1995), pp. 777–806. ISSN: 00255718, 10886842. DOI: 10.2307/2153451.

[KM05] Neal Koblitz and Alfred Menezes. "Pairing-Based Cryptography at High Security Levels (Invited Paper)". In: *10th IMA International Conference on Cryptography and Coding*. Ed. by Nigel P. Smart. Vol. 3796. LNCS. Springer, Heidelberg, Dec. 2005, pp. 13–36.

[KNP12] Kaoru Kurosawa, Ryo Nojima, and Le Trieu Phong. "Relation between Verifiable Random Functions and Convertible Undeniable Signatures, and New Constructions". In: *ACISP 12*. Ed. by Willy Susilo, Yi Mu, and Jennifer Seberry. Vol. 7372. LNCS. Springer, Heidelberg, July 2012, pp. 235–246.

[KNT18] Fuyuki Kitagawa, Ryo Nishimaki, and Keisuke Tanaka. "Obfustopia Built on Secret-Key Functional Encryption". In: *EUROCRYPT 2018, Part II*. Ed. by Jesper Buus Nielsen and Vincent Rijmen. Vol. 10821. LNCS. Springer, Heidelberg, Apr. 2018, pp. 603–648. DOI: 10.1007/978-3-319-78375-8_20.

[Koh19] Lisa Kohl. "Hunting and Gathering - Verifiable Random Functions from Standard Assumptions with Short Proofs". In: *PKC 2019, Part II*. Ed. by Dongdai Lin and Kazue Sako. Vol. 11443. LNCS. Springer, Heidelberg, Apr. 2019, pp. 408–437. DOI: 10.1007/978-3-030-17259-6_14.

[KSW08] Jonathan Katz, Amit Sahai, and Brent Waters. "Predicate Encryption Supporting Disjunctions, Polynomial Equations, and Inner Products". In: *EUROCRYPT 2008*. Ed. by Nigel P. Smart. Vol. 4965. LNCS. Springer, Heidelberg, Apr. 2008, pp. 146–162. DOI: 10.1007/978-3-540-78967-3_9.

[Lai+22] Jianchang Lai, Fuchun Guo, Willy Susilo, Peng Jiang, Guomin Yang, and Xinyi Huang. "Generic conversions from CPA to CCA without ciphertext expansion for threshold ABE with constant-size ciphertexts". In: *Information Sciences* 613 (2022), pp. 966–981. ISSN: 0020-0255. DOI: 10.1016/j.ins.2022.08.069.

[Lan23] Roman Langrehr. “On the Multi-user Security of LWE-Based NIKE”. In: *Theory of Cryptography - 21st International Conference, TCC 2023, Taipei, Taiwan, November 29 - December 2, 2023, Proceedings, Part IV*. Ed. by Guy N. Rothblum and Hoeteck Wee. Vol. 14372. Lecture Notes in Computer Science. Springer, 2023, pp. 33–62. DOI: 10.1007/978-3-031-48624-1_2.

[Laz09] Daniel Lazard. “Thirty years of Polynomial System Solving, and now?” In: *Journal of Symbolic Computation* 44.3 (2009). Polynomial System Solving in honor of Daniel Lazard, pp. 222–231. ISSN: 0747-7171. DOI: 10.1016/j.jsc.2008.03.004.

[Laz83] D. Lazard. “Gröbner bases, Gaussian elimination and resolution of systems of algebraic equations”. In: *Computer Algebra*. Ed. by J. A. van Hulzen. Berlin, Heidelberg: Springer Berlin Heidelberg, 1983, pp. 146–156. ISBN: 978-3-540-38756-5. DOI: 10.1007/3-540-12868-9_99.

[Lin16] Huijia Lin. “Indistinguishability Obfuscation from Constant-Degree Graded Encoding Schemes”. In: *EUROCRYPT 2016, Part I*. Ed. by Marc Fischlin and Jean-Sébastien Coron. Vol. 9665. LNCS. Springer, Heidelberg, May 2016, pp. 28–57. DOI: 10.1007/978-3-662-49890-3_2.

[Lin17] Huijia Lin. “Indistinguishability Obfuscation from SXDH on 5-Linear Maps and Locality-5 PRGs”. In: *CRYPTO 2017, Part I*. Ed. by Jonathan Katz and Hovav Shacham. Vol. 10401. LNCS. Springer, Heidelberg, Aug. 2017, pp. 599–629. DOI: 10.1007/978-3-319-63688-7_20.

[LLC15] Bei Liang, Hongda Li, and Jinyong Chang. “Verifiable Random Functions from (Leveled) Multilinear Maps”. In: *CANS 15*. Ed. by Michael Reiter and David Naccache. LNCS. Springer, Heidelberg, Dec. 2015, pp. 129–143. DOI: 10.1007/978-3-319-26823-1_10.

[LPR10] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. “On Ideal Lattices and Learning with Errors over Rings”. In: *EUROCRYPT 2010*. Ed. by Henri Gilbert. Vol. 6110. LNCS. Springer, Heidelberg, May 2010, pp. 1–23. DOI: 10.1007/978-3-642-13190-5_1.

[LPST16] Huijia Lin, Rafael Pass, Karn Seth, and Sidharth Telang. “Indistinguishability Obfuscation with Non-trivial Efficiency”. In: *PKC 2016, Part II*. Ed. by Chen-Mou Cheng, Kai-Min Chung, Giuseppe Persiano, and Bo-Yin Yang. Vol. 9615. LNCS. Springer, Heidelberg, Mar. 2016, pp. 447–462. DOI: 10.1007/978-3-662-49387-8_17.

[LPTWY17] Daniel Lokshtanov, Ramamohan Paturi, Suguru Tamaki, R. Ryan Williams, and Huacheng Yu. “Beating Brute Force for Systems of Polynomial Equations over Finite Fields”. In: *28th SODA*. Ed. by Philip N. Klein. ACM-SIAM, Jan. 2017, pp. 2190–2202. DOI: 10.1137/1.9781611974782.143.

[LS15] Adeline Langlois and Damien Stehlé. "Worst-case to average-case reductions for module lattices". In: *Des. Codes Cryptogr.* 75.3 (2015), pp. 565–599. DOI: 10.1007/s10623-014-9938-4.

[LSSS17] Benoît Libert, Amin Sakzad, Damien Stehlé, and Ron Steinfeld. "All-But-Many Lossy Trapdoor Functions and Selective Opening Chosen-Ciphertext Security from LWE". In: *CRYPTO 2017, Part III*. Ed. by Jonathan Katz and Hovav Shacham. Vol. 10403. LNCS. Springer, Heidelberg, Aug. 2017, pp. 332–364. DOI: 10.1007/978-3-319-63697-9_12.

[LT17] Huijia Lin and Stefano Tessaro. "Indistinguishability Obfuscation from Trilinear Maps and Block-Wise Local PRGs". In: *CRYPTO 2017, Part I*. Ed. by Jonathan Katz and Hovav Shacham. Vol. 10401. LNCS. Springer, Heidelberg, Aug. 2017, pp. 630–660. DOI: 10.1007/978-3-319-63688-7_21.

[LV16] Huijia Lin and Vinod Vaikuntanathan. "Indistinguishability Obfuscation from DDH-Like Assumptions on Constant-Degree Graded Encodings". In: *57th FOCS*. Ed. by Irit Dinur. IEEE Computer Society Press, Oct. 2016, pp. 11–20. DOI: 10.1109/FOCS.2016.11.

[LW14] Allison B. Lewko and Brent Waters. "Why Proving HIBE Systems Secure Is Difficult". In: *EUROCRYPT 2014*. Ed. by Phong Q. Nguyen and Elisabeth Oswald. Vol. 8441. LNCS. Springer, Heidelberg, May 2014, pp. 58–76. DOI: 10.1007/978-3-642-55220-5_4.

[Lys02] Anna Lysyanskaya. "Unique Signatures and Verifiable Random Functions from the DH-DDH Separation". In: *CRYPTO 2002*. Ed. by Moti Yung. Vol. 2442. LNCS. Springer, Heidelberg, Aug. 2002, pp. 597–612. DOI: 10.1007/3-540-45708-9_38.

[Mac02] Francis Sowerby Macaulay. "Some Formulae in Elimination". In: *Proceedings of the London Mathematical Society* s1-35.1 (1902), pp. 3–27. DOI: 10.1112/plms/s1-35.1.3.

[Mac16] Francis Sowerby Macaulay. "The algebraic theory of modular systems". In: *Cambridge Mathematical Library* xxxi (1916).

[Mau02] Ueli M. Maurer. "Indistinguishability of Random Systems". In: *EUROCRYPT 2002*. Ed. by Lars R. Knudsen. Vol. 2332. LNCS. Springer, Heidelberg, Apr. 2002, pp. 110–132. DOI: 10.1007/3-540-46035-7_8.

[Mau05] Ueli M. Maurer. "Abstract Models of Computation in Cryptography (Invited Paper)". In: *10th IMA International Conference on Cryptography and Coding*. Ed. by Nigel P. Smart. Vol. 3796. LNCS. Springer, Heidelberg, Dec. 2005, pp. 1–12.

[Mil82] F. Miller. *Telegraphic Code to Insure Privacy and Secrecy in the Transmission of Telegrams*. C.M. Cornwell, 1882. URL: https://books.google.ch/books?id=tT9WAAAAYAAJ.

[MM84] H. Michael Möller and Ferdinando Mora. “Upper and lower bounds for the degree of Groebner bases”. In: *EUROSAM 84*. Ed. by John Fitch. Berlin, Heidelberg: Springer Berlin Heidelberg, 1984, pp. 172–183. ISBN: 978-3-540-38893-7. DOI: 10.1007/BFb0032840.

[MMDB08] Mohamed Saied Emam Mohamed, Wael Said Abd Elmageed Mohamed, Jintai Ding, and Johannes A. Buchmann. “MXL2: Solving Polynomial Equations over GF(2) Using an Improved Mutant Strategy”. In: *Post-quantum cryptography, second international workshop, PQCRYPTO 2008*. Ed. by Johannes Buchmann and Jintai Ding. Springer, Heidelberg, Oct. 2008, pp. 203–215. DOI: 10.1007/978-3-540-88403-3_14.

[MP12] Daniele Micciancio and Chris Peikert. “Trapdoors for Lattices: Simpler, Tighter, Faster, Smaller”. In: *EUROCRYPT 2012*. Ed. by David Pointcheval and Thomas Johansson. Vol. 7237. LNCS. Springer, Heidelberg, Apr. 2012, pp. 700–718. DOI: 10.1007/978-3-642-29011-4_41.

[MRV99] Silvio Micali, Michael O. Rabin, and Salil P. Vadhan. “Verifiable Random Functions”. In: *40th FOCS*. IEEE Computer Society Press, Oct. 1999, pp. 120–130. DOI: 10.1109/SFFCS.1999.814584.

[MST03] Elchanan Mossel, Amir Shpilka, and Luca Trevisan. “On e-Biased Generators in NC0”. In: *44th FOCS*. IEEE Computer Society Press, Oct. 2003, pp. 136–145. DOI: 10.1109/SFCS.2003.1238188.

[MSZ16] Eric Miles, Amit Sahai, and Mark Zhandry. “Annihilation Attacks for Multilinear Maps: Cryptanalysis of Indistinguishability Obfuscation over GGH13”. In: *CRYPTO 2016, Part II*. Ed. by Matthew Robshaw and Jonathan Katz. Vol. 9815. LNCS. Springer, Heidelberg, Aug. 2016, pp. 629–658. DOI: 10.1007/978-3-662-53008-5_22.

[MVO91] Alfred Menezes, Scott A. Vanstone, and Tatsuaki Okamoto. “Reducing Elliptic Curve Logarithms to Logarithms in a Finite Field”. In: *23rd ACM STOC*. ACM Press, May 1991, pp. 80–89. DOI: 10.1145/103418.103434.

[Nie21] David Niehues. “Verifiable Random Functions with Optimal Tightness”. In: *PKC 2021, Part II*. Ed. by Juan Garay. Vol. 12711. LNCS. Springer, Heidelberg, May 2021, pp. 61–91. DOI: 10.1007/978-3-030-75248-4_3.

[Olv06] Peter J. Olver. “On Multivariate Interpolation”. In: *Studies in Applied Mathematics* 116.2 (2006), pp. 201–240. DOI: 10.1111/j.1467-9590.2006.00335.x.

[ONe10] Adam O’Neill. *Definitional Issues in Functional Encryption*. Cryptology ePrint Archive, Report 2010/556. https://eprint.iacr.org/2010/556. 2010.

[Pie12] Krzysztof Pietrzak. “Cryptography from Learning Parity with Noise”. In: *SOFSEM 2012: Theory and Practice of Computer Science*. Ed. by Mária Bieliková, Gerhard Friedrich, Georg Gottlob, Stefan Katzenbeisser, and György Turán. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 99–114.

[PST14] Rafael Pass, Karn Seth, and Sidharth Telang. “Indistinguishability Obfuscation from Semantically-Secure Multilinear Encodings”. In: *CRYPTO 2014, Part I*. Ed. by Juan A. Garay and Rosario Gennaro. Vol. 8616. LNCS. Springer, Heidelberg, Aug. 2014, pp. 500–517. DOI: 10.1007/978-3-662-44371-2_28.

[Rab80] Michael O. Rabin. “Probabilistic Algorithms in Finite Fields”. In: *SIAM Journal on Computing* 9.2 (1980), pp. 273–280. DOI: 10.1137/0209024.

[Reg05] Oded Regev. “On lattices, learning with errors, random linear codes, and cryptography”. In: *37th ACM STOC*. Ed. by Harold N. Gabow and Ronald Fagin. ACM Press, May 2005, pp. 84–93. DOI: 10.1145/1060590.1060603.

[Ros18] Razvan Rosie. “Adaptive-Secure VRFs with Shorter Keys from Static Assumptions”. In: *CANS 18*. Ed. by Jan Camenisch and Panos Papadimitratos. Vol. 11124. LNCS. Springer, Heidelberg, Sept. 2018, pp. 440–459. DOI: 10.1007/978-3-030-00434-7_22.

[RSA78] R. L. Rivest, A. Shamir, and L. Adleman. “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems”. In: *Commun. ACM* 21.2 (Feb. 1978), pp. 120–126. ISSN: 0001-0782. DOI: 10.1145/359340.359342.

[Sch80] J. T. Schwartz. “Fast Probabilistic Algorithms for Verification of Polynomial Identities”. In: *J. ACM* 27.4 (Oct. 1980), pp. 701–717. ISSN: 0004-5411. DOI: 10.1145/322217.322225.

[Sha49] C. E. Shannon. “Communication theory of secrecy systems”. In: *The Bell System Technical Journal* 28.4 (1949), pp. 656–715. DOI: 10.1002/j.1538-7305.1949.tb00928.x.

[Sha84] Adi Shamir. “Identity-Based Cryptosystems and Signature Schemes”. In: *CRYPTO’84*. Ed. by G. R. Blakley and David Chaum. Vol. 196. LNCS. Springer, Heidelberg, Aug. 1984, pp. 47–53.

[Sho88] Victor Shoup. “New Algorithms for Finding Irreducible Polynomials over Finite Fields”. In: *29th FOCS*. IEEE Computer Society Press, Oct. 1988, pp. 283–290. DOI: 10.1109/SFCS.1988.21944.

[Sho97] Victor Shoup. “Lower Bounds for Discrete Logarithms and Related Problems”. In: *EUROCRYPT’97*. Ed. by Walter Fumy. Vol. 1233. LNCS. Springer, Heidelberg, May 1997, pp. 256–266. DOI: 10.1007/3-540-69053-0_18.

[SS10] Amit Sahai and Hakan Seyalioglu. "Worry-free encryption: functional encryption with public keys". In: *ACM CCS 2010.* Ed. by Ehab Al-Shaer, Angelos D. Keromytis, and Vitaly Shmatikov. ACM Press, Oct. 2010, pp. 463–472. DOI: `10.1145/1866307.1866359`.

[SSW09] Emily Shen, Elaine Shi, and Brent Waters. "Predicate Privacy in Encryption Systems". In: *TCC 2009.* Ed. by Omer Reingold. Vol. 5444. LNCS. Springer, Heidelberg, Mar. 2009, pp. 457–473. DOI: `10.1007/978-3-642-00457-5_27`.

[SW05] Amit Sahai and Brent R. Waters. "Fuzzy Identity-Based Encryption". In: *EUROCRYPT 2005.* Ed. by Ronald Cramer. Vol. 3494. LNCS. Springer, Heidelberg, May 2005, pp. 457–473. DOI: `10.1007/11426639_27`.

[SW14] Amit Sahai and Brent Waters. "How to use indistinguishability obfuscation: deniable encryption, and more". In: *46th ACM STOC.* Ed. by David B. Shmoys. ACM Press, May 2014, pp. 475–484. DOI: `10.1145/2591796.2591825`.

[TÜ23] Erkan Tairi and Akin Ünal. *Lower Bounds for Lattice-based Compact Functional Encryption.* Cryptology ePrint Archive, Paper 2023/719. May 2023. URL: `https://eprint.iacr.org/2023/719`.

[TW10] Enrico Thomae and Christopher Wolf. *Solving Systems of Multivariate Quadratic Equations over Finite Fields or: From Relinearization to MutantXL.* Cryptology ePrint Archive, Report 2010/596. `https://eprint.iacr.org/2010/596`. 2010.

[Üna20] Akin Ünal. "Impossibility Results for Lattice-Based Functional Encryption Schemes". In: *EUROCRYPT 2020, Part I.* Ed. by Anne Canteaut and Yuval Ishai. Vol. 12105. LNCS. Springer, Heidelberg, May 2020, pp. 169–199. DOI: `10.1007/978-3-030-45721-1_7`.

[Üna23a] Akin Ünal. *New Baselines for Local Pseudorandom Number Generators by Field Extensions.* Cryptology ePrint Archive, Paper 2023/550. Apr. 2023. URL: `https://eprint.iacr.org/2023/550`.

[Üna23b] Akin Ünal. *Worst-Case Subexponential Attacks on PRGs of Constant Degree or Constant Locality.* Cryptology ePrint Archive, Report 2023/119. `https://eprint.iacr.org/2023/119`. 2023.

[Üna23c] Akin Ünal. "Worst-Case Subexponential Attacks on PRGs of Constant Degree or Constant Locality". In: *EUROCRYPT 2023, Part I.* Ed. by Carmit Hazay and Martijn Stam. Vol. 14004. LNCS. Springer, Heidelberg, Apr. 2023, pp. 25–54. DOI: `10.1007/978-3-031-30545-0_2`.

[Ver01] Eric R. Verheul. "Evidence that XTR Is More Secure than Supersingular Elliptic Curve Cryptosystems". In: *EUROCRYPT 2001.* Ed. by Birgit Pfitzmann. Vol. 2045. LNCS. Springer, Heidelberg, May 2001, pp. 195–210. DOI: `10.1007/3-540-44987-6_13`.

[Ver26] G. S. Vernam. "Cipher Printing Telegraph Systems For Secret Wire and Radio Telegraphic Communications". In: *Transactions of the American Institute of Electrical Engineers* XLV (1926), pp. 295–301. DOI: 10.1109/T-AIEE.1926.5061224.

[Vil97] Gilles Villard. *A Study of Coppersmith's Block Wiedemann Algorithm Using Matrix Polynomials*. Rapport de recherche / [IMAG]. IMAG, Institut d'informatique et de mathématiques appliquées de Grenoble, 1997. URL: https://perso.ens-lyon.fr/gilles.villard/BIBLIOGRAPHIE/PDF/rr0497.pdf.

[Wat15] Brent Waters. "A Punctured Programming Approach to Adaptively Secure Functional Encryption". In: *CRYPTO 2015, Part II*. Ed. by Rosario Gennaro and Matthew J. B. Robshaw. Vol. 9216. LNCS. Springer, Heidelberg, Aug. 2015, pp. 678–697. DOI: 10.1007/978-3-662-48000-7_33.

[Wee20] Hoeteck Wee. "Functional Encryption for Quadratic Functions from k-Lin, Revisited". In: *TCC 2020, Part I*. Ed. by Rafael Pass and Krzysztof Pietrzak. Vol. 12550. LNCS. Springer, Heidelberg, Nov. 2020, pp. 210–228. DOI: 10.1007/978-3-030-64375-1_8.

[Wie86] Douglas Wiedemann. "Solving sparse linear equations over finite fields". In: *IEEE Transactions on Information Theory* 32.1 (1986), pp. 54–62. DOI: 10.1109/TIT.1986.1057137.

[Wil14] Richard Ryan Williams. "The Polynomial Method in Circuit Complexity Applied to Algorithm Design (Invited Talk)". In: *34th International Conference on Foundation of Software Technology and Theoretical Computer Science (FSTTCS 2014)*. Ed. by Venkatesh Raman and S. P. Suresh. Vol. 29. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2014, pp. 47–60. DOI: 10.4230/LIPIcs.FSTTCS.2014.47.

[Woo96] Trevor D. Wooley. "A Note on Simultaneous Congruences". In: *Journal of Number Theory* 58.2 (1996), pp. 288–297. ISSN: 0022-314X. DOI: 10.1006/jnth.1996.0078.

[Yam17] Shota Yamada. "Asymptotically Compact Adaptively Secure Lattice IBEs and Verifiable Random Functions via Generalized Partitioning Techniques". In: *CRYPTO 2017, Part III*. Ed. by Jonathan Katz and Hovav Shacham. Vol. 10403. LNCS. Springer, Heidelberg, Aug. 2017, pp. 161–193. DOI: 10.1007/978-3-319-63697-9_6.

[YC04] Bo-Yin Yang and Jiun-Ming Chen. "Theoretical Analysis of XL over Small Fields". In: *ACISP 04*. Ed. by Huaxiong Wang, Josef Pieprzyk, and Vijay Varadharajan. Vol. 3108. LNCS. Springer, Heidelberg, July 2004, pp. 277–288. DOI: 10.1007/978-3-540-27800-9_24.

[YC05] Bo-Yin Yang and Jiun-Ming Chen. "All in the XL Family: Theory and Practice". In: *ICISC 04*. Ed. by Choonsik Park and Seongtaek Chee. Vol. 3506. LNCS. Springer, Heidelberg, Dec. 2005, pp. 67–86.

[YCY22] Li Yao, Yilei Chen, and Yu Yu. "Cryptanalysis of Candidate Obfuscators for Affine Determinant Programs". In: *EUROCRYPT 2022, Part I*. Ed. by Orr Dunkelman and Stefan Dziembowski. Vol. 13275. LNCS. Springer, Heidelberg, May 2022, pp. 645–669. DOI: 10.1007/978-3-031-06944-4_22.

[YDHTS15] Takanori Yasuda, Xavier Dahan, Yun-Ju Huang, Tsuyoshi Takagi, and Kouichi Sakurai. *MQ Challenge: Hardness Evaluation of Solving Multivariate Quadratic Problems*. Cryptology ePrint Archive, Report 2015/275. https://eprint.iacr.org/2015/275. 2015.

[Zha22] Mark Zhandry. "To Label, or Not To Label (in Generic Groups)". In: *CRYPTO 2022, Part III*. Ed. by Yevgeniy Dodis and Thomas Shrimpton. Vol. 13509. LNCS. Springer, Heidelberg, Aug. 2022, pp. 66–96. DOI: 10.1007/978-3-031-15982-4_3.

[Zic17] Lior Zichron. *Locally Computable Arithmetic Pseudorandom Generators*. 2017. URL: https://www.bennyapplebaum.sites.tau.ac.il/_files/ugd/f706bf_501515c9cd7744c498935684bd1648a2.pdf.

[Zip79] Richard Zippel. "Probabilistic algorithms for sparse polynomials". In: *Symbolic and Algebraic Computation*. Ed. by Edward W. Ng. Berlin, Heidelberg: Springer Berlin Heidelberg, 1979, pp. 216–226. ISBN: 978-3-540-35128-3. DOI: 10.1007/3-540-09519-5_73.

[ZZK22] Cong Zhang, Hong-Sheng Zhou, and Jonathan Katz. "An Analysis of the Algebraic Group Model". In: *ASIACRYPT 2022, Part IV*. Ed. by Shweta Agrawal and Dongdai Lin. Vol. 13794. LNCS. Springer, Heidelberg, Dec. 2022, pp. 310–322. DOI: 10.1007/978-3-031-22972-5_11.

Index