Impossibility Results for Lattice-Based Functional Encryption Schemes

Akın Ünal*
ETH Zurich, Switzerland
Work done while the author was at Karlsruhe Institute of Technology.

April 30, 2023

Abstract

Functional Encryption denotes a form of encryption where a master secret key-holder can control which functions a user can evaluate on encrypted data. Learning With Errors (LWE) (Regev, STOC'05) is known to be a useful cryptographic hardness assumption which implies strong primitives such as, for example, fully homomorphic encryption (Brakerski-Vaikuntanathan, FOCS'11) and lockable obfuscation (Goyal et al., Wichs et al., FOCS'17). Despite its strength, however, there is just a limited number of functional encryption schemes which can be based on LWE. In fact, there are functional encryption schemes which can be achieved by using pairings but for which no secure instantiations from lattice-based assumptions are known: function-hiding inner product encryption (Lin, Baltico et al., CRYPTO'17) and compact quadratic functional encryption (Abdalla et al., CRYPTO'18). This raises the question whether there are some mathematical barriers which hinder us from realizing function-hiding and compact functional encryption schemes from lattice-based assumptions as LWE.

To study this problem, we prove an impossibility result for function-hiding functional encryption schemes which meet some algebraic restrictions at ciphertext encryption and decryption. Those restrictions are met by a lot of attribute-based, identity-based and functional encryption schemes whose security stems from LWE. Therefore, we see our results as important indications why it is hard to construct new functional encryption schemes from LWE and which mathematical restrictions have to be overcome to construct secure lattice-based functional encryption schemes for new functionalities.

Keywords: Functional Encryption, Function-Hiding, Impossibility, LWE, Lattice-based, Online/Offline.

*The author is supported by ERC Project 724307 'PREP-CRYPTO'.

1 Introduction

Functional Encryption (FE) schemes are special encryption schemes in which the holder of a master secret key can issue secret keys for specific functions to users. By knowing a secret key for a function f and a ciphertext for a message x, an adversary shall learn nothing more of x than $f(x)$. FE schemes have proven to be extremely versatile. Not only does their notion generalize other forms of encryption like Attribute-Based (ABE) or Identity-Based Encryption (IBE), but also do we know that compact single-key FE and linearly compact FE for cubic polynomials together with plausible assumptions imply indistinguishability obfuscation [AJ15, BV15, LT17].

Function-Hiding Functional Encryption (FHFE) schemes are an even stronger subclass of FE where we demand 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 of $f(x)$; i.e., the secret keys now hide the functions they are supposed to evaluate.

We know that FE schemes *with a bounded number of secret keys*, an adversary may learn, are already achievable from minimal assumptions [AV19]. However, if we try to achieve security for an unbounded number of secret keys, then we are left with (function-hiding) inner-product encryption, linearly compact quadratic FE and FE schemes for constant-degree polynomials which are yielded by relinearizing. Of course, there are special cases of FE like attribute-based and identity-based encryption schemes. In those schemes, a ciphertext is accompanied with a non-hidden attribute or identity and decryption is successful iff the attribute/identity matches the policy of the secret key. However, the main focus in this work are FE schemes, since we are interested in schemes which perform various computations on hidden inputs. We stress here that for linearly compact quadratic FE and function-hiding inner-product FE there are just pairing-based constructions known so far [BJK15, DDM16, BCFG17, Lin17, ACF$^+$18, Gay20].

Learning With Errors (LWE) [Reg05] is a well-established hardness assumption. It states that it is hard to solve a system of linear equations over a modulus q, if the solution has sufficient entropy, the coefficients of the equations are chosen uniformly random from $\mathbb{Z}_q$ and one column of the presented system has been perturbed by a small noise-vector whose entries are sampled from a suitable error-distribution. Because of its strong homomorphic properties, there are fully homomorphic encryption schemes and lockable obfuscation schemes whose security can be proven solely under LWE [BGV12, GKW17, WZ17]. Up to now, it is not possible to construct those schemes from other standard assumptions. Intuitively, one would assume that its homomorphic properties imply a lot of different FE schemes. But as we have stressed, the most complex already existing FE schemes cannot be replicated by lattice-based constructions. In fact, inner product encryption is the only FE scheme whose security can be based on LWE (again, putting ABE and IBE aside). Because of the aforementioned amply homomorphic properties of LWE, this is very surprising and leads us to the following question:

What hinders us from constructing function-hiding inner-product encryption schemes whose security can be proven solely from the learning with errors assumption?

We show that there are two properties, both very common under LWE-based FE schemes, which make it impossible for a function-hiding inner-product encryption scheme to be secure. The first property lies in the decryption algorithms of LWE-based encryption schemes: If we take a close look at the pairing-based schemes, we see that decryption is always complex, for it involves computing discrete logarithms of the target group of the pairing. On the other hand, a lot of LWE-based IBE and FE schemes have simple decryption algorithms [CHKP10, ABB10, AFV11, BRS13, ABDP15, ALS16]. In most cases, for moduli $q > p > 1$, a secret key sk in such a scheme usually determines a multivariate polynomial $g_{\mathsf{sk}}(Y_1, \ldots, Y_s)$ of constant total degree, while the ciphertext is a vector $\mathsf{ct} \in \mathbb{Z}_q^s$. At decryption, the polynomial is evaluated at the ciphertext which yields a value $g_{\mathsf{sk}}(\mathsf{ct}) \in \mathbb{Z}_q$; this value will be *rounded to the nearest number of* $\mathbb{Z}_p$, i.e., it will be divided by $\lfloor q/p \rfloor$ and then rounded to the nearest integer in $\{0, \ldots, p-1\}$. In full detail, this means

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

We believe that this property already suffices to render a FHFE scheme insecure. Therefore, we state here the following conjecture:

Conjecture 1. Let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be a correct private-key functional encryption scheme for computing inner-products of vectors in $\mathbb{Z}_p^n$. If there is a constant $d' \in \mathbb{N}$ and a polynomial s in the security parameter, s.t.

- each ciphertext ct sampled by Enc is a vector in $\mathbb{Z}_q^s$,
- each secret key sk sampled by KeyGen is a multivariate polynomial in $\mathbb{Z}_q[Y_1, \ldots, Y_s]$ of total degree $\leq d'$
- and the decryption algorithm works by

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

then FE cannot be function-hiding secure for an unbounded number of secret keys.

We leave it as an open question to prove or refute conjecture 1. Instead, we prove in this work a weaker version of the above statement.

If we are to take a closer look at the aforementioned IBE and FE schemes and some ABE schemes [GVW13, BGG+14], we can distinguish an additional property which seems to be common for some LWE-based schemes. They tend to have very *algebraic* encryption algorithms. Take, for example, a closer look at ciphertext encryption in the LWE-based inner-product encryption schemes of Agrawal et al. [ALS16]. For an input vector $x \in \{0, \ldots, p-1\}^l$ and two publicly known matrices $A \in \mathbb{Z}_q^{m \times n}, U \in \mathbb{Z}_q^{l \times n}$, ciphertexts are generated by sampling a uniformly random vector $s \leftarrow \mathbb{Z}_q^n$, two gaussian noise vectors $e_0 \leftarrow \mathcal{D}_{\mathbb{Z}^m, \alpha q}, e_1 \leftarrow \mathcal{D}_{\mathbb{Z}^l, \alpha q}$ and outputting

$$\mathsf{ct} = (As + e_0, Us + e_1 + b \cdot x)$$

where b is either $\lfloor q/K \rfloor$ or p^{k-1}. Note that we can distinguish two parts in this encryption algorithm:

– a very complex *offline* part, where $m + l$ multivariate degree-1 polynomials

$$g_1(X), \ldots, g_m(X), h_1(X), \ldots, h_l(X)$$

are sampled by only knowing the public key (A, U, p, q, K) and without looking at the input x:

$$g_i(X_1, \ldots, X_l) := \langle a_i \mid s \rangle + e_{0,i},$$
$$h_i(X_1, \ldots, X_l) := \langle u_i \mid s \rangle + e_{1,i} + \lfloor q/K \rfloor \cdot X_i$$

– and a simple *online* part which just consists of inserting x in the polynomials sampled before and outputting the ciphertext

$$\mathsf{ct} = (g_1(x), \ldots, g_m(x), h_1(x), \ldots, h_l(x)).$$

This distinction in a complex offline and a simple online part can be seen in the other aforementioned schemes, too. Therefore, we extract it as an additional characteristic of some LWE-based schemes and make it more precise in the following:

We say Enc is an encryption algorithm of *depth* d over $\mathbb{Z}_q$, if there is a ppt algorithm $\mathsf{Enc}_{\text{offline}}$, s.t. we have for each master secret key msk and input $x \in \mathbb{Z}_p^n$:

$$\begin{aligned}
\mathsf{Enc}(\mathsf{msk}, x) = \{ & \\
& (r_1, \ldots, r_s) \leftarrow \mathsf{Enc}_{\text{offline}}(\mathsf{msk}) && (1) \\
& \mathsf{return}\ (r_1(x), \ldots, r_s(x)) && (2) \\
\} &
\end{aligned}$$

where we demand that each r_i is a multivariate polynomial in $\mathbb{Z}_q[X_1, \ldots, X_n]$ of total degree $\leq d$. We will call line (1) the *offline* part and line (2) the *online* part of Enc. Indeed, with this additional property we can prove an FHFE scheme to be insecure.

1.1 Contribution

For moduli $q = q(\lambda) > p = p(\lambda)$ such that q is prime, $\frac{q}{p}$ is polynomially bounded and p is not bounded by a constant, we prove the following:

Theorem 1 (Informal Main Theorem)**.** *Assume that the prerequisites of conjecture 1 hold and that additionally* Enc *is of* depth d *over* $\mathbb{Z}_q$ *for some constant* $d \in \mathbb{N}$*.*

Then, FE *cannot be function-hiding secure for an unbounded number of secret keys.*

To be more precise, we give a bound of the maximum number of secret keys which can be issued to an adversary before he can break FE (corollary 4). On a very high level, our proof idea is to use the algebraic structure of the composition $\mathsf{Dec} \circ \mathsf{Enc}$. By doing so, we show that the decryption noises are generated in a very algebraic way, are small and contain information about the encrypted ciphertexts. Therefore, we can prove theorem 1 by analysing them.

As an additional result, we show that private-key encryption schemes where the encryption algorithms are of constant depth and the ciphertext vectors are short enough cannot be secure (theorem 6 and corollary 3). This result does not depend on the decryption algorithms of the private-key encryption schemes.

1.1.1 Generality of Our Results.

We note here that there are a lot of LWE-based ABE schemes whose decryption algorithms are too complex to be subsumed by the equation

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = \left\lceil \frac{\mathsf{sk}(\mathsf{ct})}{\lfloor q/p \rfloor} \right\rfloor \tag{3}$$

for a constant degree polynomial sk. This is because they allow policy-predicates which cannot be computed by constant-depth circuits. Since the policy-predicate needs to be computed at decryption, their decryption algorithms must be at least as complicated as the most complex policy-predicate they allow. However, the aforementioned ABE schemes in [GVW13, BGG$^+$14] have decryption algorithms that become simple enough to fit equation (3), if we restrict the policy-circuits in those schemes to be of constant depth and if attributes and policy match at decryption.

1.1.2 Two-Input Quadratic Functional Encryption.

We can derive from theorem 1 an impossibility result for 2-input quadratic FE schemes. A 2-input quadratic FE scheme evaluates functions with two distinguished inputs and has a left and a right encryption algorithm. To decrypt a value $f(x, y)$, one needs a secret key for f, a left ciphertext for x and a right ciphertext for y. Since such a scheme contains a secret key for the quadratic function $f(x, y) = \langle x \mid y \rangle$, it can emulate a function-hiding inner-product encryption scheme, even if it is only single-key secure.

Corollary 1. *Let* $2\,\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}^R, \mathsf{Enc}^L, \mathsf{Dec})$ *be a correct private-key 2-input functional encryption scheme for quadratic functions* $f : \mathbb{Z}_p^n \times \mathbb{Z}_p^n \to \mathbb{Z}_p$*. If there are* $s \in \mathsf{poly}(\lambda)$ *and a constant* $d' \in \mathbb{N}$*, s.t.*

- Enc^L *is of constant depth* d *over* $\mathbb{Z}_q$*,*
- *each ciphertext* ct^L *sampled by* Enc^L *is a vector in* $\mathbb{Z}_q^s$*,*
- *each pair of a secret key* sk *and a right ciphertext* ct^R *determines a multivariate polynomial* $g_{\mathsf{sk},\mathsf{ct}^R} \in \mathbb{Z}_q[X_1, \ldots, X_s]$ *of total degree* $\leq d'$ *s.t. the decryption algorithm works by*

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}^L, \mathsf{ct}^R) = \left\lceil \frac{g_{\mathsf{sk},\mathsf{ct}^R}(\mathsf{ct})}{\lfloor q/p \rfloor} \right\rfloor,$$

then $2\,\mathsf{FE}$ *cannot be single-key secure.*

1.2 Interpretation, Limitations and Open Problems

Parameter Restrictions To prove theorem 1, we assume that the exterior modulus q of the FHFE scheme FE is prime. Furthermore, we need that the fraction q/p is bounded by a polynomial in the security parameter λ and that the interior modulus p is for almost all λ greater than some constant which depends on the depth of FE. Note that q/p is usually a bound for the error noise used in LWE-based schemes. Since LWE is assumed to be hard, even if its modulus q is a prime and the deviation of its error noise is bounded by a polynomial in λ, we do not think that those requirements are big restrictions for our results.

Interpretation and Open Problems We see the results in this paper as a useful argument in understanding the difficulties in constructing LWE-based function-hiding functional encryption schemes. An even more useful argument would be to close the gap and prove conjecture 1. Because of theorem 1, to prove our conjecture, it now suffices to transform a function-hiding inner-product encryption scheme which is correct and secure and fulfils the requirements of the conjecture to one that fulfils the requirements of theorem 1. In other words, it suffices to take an FHFE scheme which already decrypts in an LWE-like manner and simplify its encryption algorithm to one of constant depth which stays secure and correct.

Another way to extend the results here is to prove theorem 1 for encryption algorithms where, in the online part, one first computes a bit-decomposition $G^{-1}(x)$ of an input vector x and then applies the polynomials sampled in the offline part to $G^{-1}(x)$. A lot of the techniques here would not be suitable for this task; indeed, one would need to develop more advanced techniques to show this.

A Note on Ring LWE One can ask himself, if RLWE, a more algebraic version of LWE introduced in [LPR10], can help to overcome the requirements of theorem 1. We want to point out that, as long as a FHFE scheme meets the requirements of theorem 1, it does not matter, if its apparent security stems from LWE or RLWE. Let us explain this in more detail:

Let q, p be like in theorem 1. For some $m \in \mathsf{poly}(\lambda)$, define the ring

$$R_q := \mathbb{Z}[X]/(q, X^m + 1).$$

Further, let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be a functional encryption scheme which fulfils the requirements of theorem 1, if one replaces $\mathbb{Z}_q$ by R_q. I.e., FE shall fulfil the following requirements:

- there is an $s \in \mathsf{poly}(\lambda)$ s.t. each ciphertext $\mathsf{ct} \in \mathsf{supp}(\mathsf{Enc})$ is an element of R_q^s,
- each secret key sk sampled KeyGen is a multivariate polynomial $R_q[Z_1, \ldots, Z_s]$of total degree $\leq d_2$,
- the decryption algorithm works by

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = \left\lceil \frac{\mathsf{sk}(\mathsf{ct})}{\lfloor q/p \rfloor} \right\rfloor \in R_p := \mathbb{Z}[X]/(p, X^m + 1)$$

- and Enc is of constant depth d_1 over R_q, i.e., there is a ppt algorithm $\mathsf{Enc}_{\text{offline}}$ which on input msk outputs s multivariate polynomials in $R_q[Y_1, \ldots, Y_n]$ of total degree $\leq d_1$ s.t. Enc works by

$$\begin{aligned}\mathsf{Enc}(\mathsf{msk}, x) = \{ & \\ & (r_1, \ldots, r_s) \leftarrow \mathsf{Enc}_{\text{offline}}(\mathsf{msk}) \\ & \mathsf{return}\ (r_1(x), \ldots, r_s(x)) \\ \}.\end{aligned}$$

Then, one can show that FE cannot be function-hiding secure for an unbounded number of secret-keys. This can be done by converting FE to a functional encryption scheme which meets the requirements of theorem 1 (over $\mathbb{Z}_q$). To this end, it suffices to note, that each ciphertext $\mathsf{ct} \in$

$\mathsf{supp}(\mathsf{Enc})$ can be interpreted as vector in $\mathbb{Z}_q^{s \cdot m} \cong R_q^s$ and that each polynomial $g \in R_q[Y_1, \ldots, Y_s]$ of degree d can be interpreted as a tuple of m polynomials

$$(g_1, \ldots, g_m) \in \mathbb{Z}_q[Y'_{1,1}, \ldots, Y'_{1,m}, \ldots, Y'_{s,1}, \ldots, Y'_{s,m}]$$

each of total degree $\leq d$.

1.3 Related Work

The idea of decomposing encryption algorithms into simple online and complex offline parts has already been studied with the purpose of finding FE schemes with practical usages (we cite [HW14, AR17] as examples). However, to the best of our knowledge, this is the first work where the online/offline structure of encryption has been used to prove an impossibility result.

Ananth and Vaikuntanathan showed that FE for $\mathsf{P/poly}$ with a bounded number of secret keys can already be achieved from minimal assumptions, i.e. public-key encryption in the asymmetric setting and one-way functions in the symmetric setting [AV19]. The ciphertexts in their schemes are growing linearly with the number of secret keys which can be handed out to an adversary. It is presumably hard to improve their result, since we know that a bounded FE scheme with sufficiently compact ciphertexts would already imply indistinguishability obfuscation [AJ15, BV15].

As mentioned, it is hard to construct FE schemes for stronger functionalities. In recent years, researchers circumvented this problem and looked at novel FE schemes with additional properties: Abdalla, Chotard and other researchers constructed mulit-input and decentralized multi-client inner-product encryption schemes [ACF+18, CDG+18, ABKW19, ACF+19]. Those are inner-product encryption schemes where a function has multiple inputs and to decrypt one needs a secret key and multiple suitable ciphertexts. In the decentralized schemes, one gets rid of the master secret key holder. Jain et al. introduced the notion of 3-restricted FE [AJS18, JLMS19], which can be understood as cubic FE where a ciphertext just hides two out of three factors.

1.4 Technical Overview

To prove theorem 1, we need to show the existence of a selective adversary who wins the function-hiding IND-CPA game against the function-hiding inner-product encryption scheme FE. In this game, the adversary submits an unbounded number of inputs x_i^0 and functions f_j^0 for world 0 and an unbounded number of inputs x_i^1 and functions f_j^1 for world 1. Then, the challenger draws a random bit $b \leftarrow \{0, 1\}$ and sends the corresponding ciphertexts and secret keys of world b to the adversary. The adversary wins, if he guesses b correctly and if the submitted inputs and functions would not tell him trivially in which world he lives, i.e., if we have for all i and j

$$f_j^0(x_i^0) = f_j^1(x_i^1).$$

We do not directly construct an adversary to break FE. Instead, we show how an adversary can reduce the problem of breaking FE to the problem of breaking other encryption schemes with additional properties. To do so, we apply multiple transformations to FE. Eventually, we end with a private-key encryption scheme whose ciphertexts are short integer vectors and whose encryption algorithm is of constant depth. Then, we construct a simple adversary who can break such encryption schemes.

To make our argument go through, we need the transformations to preserve the security and correctness of the transformed schemes. It is easy to see that security is preserved, since we ensure that all changes to FE can be computed by an adversary while he plays the above security game against FE. On the other hand, we can not always guarantee that our transformations preserve correctness. In fact, one transformation step applied to FE changes it in such a way that decryption succeeds only in a non-negligible number of cases. Furthermore, it is important that at each time we have an encryption algorithm of constant depth. This means, each transformation step either changes the encryption algorithm without changing its depth or at most changes its depth to another constant value.

Our proof consists of three major steps:

(1) We first change FE s.t. all ciphertexts have short entries relative to the modulus q. To do this, the adversary queries a lot of secret keys for the zero-function and learns, by doing so, the structure of the space of secret keys. Then, he can exchange a ciphertext with a vector of decryption noises. Those noises have to be short, because otherwise they would make a correct decryption impossible. On the other hand, however, we show that those noises contain enough information about the original ciphertext to make decryption possible in a non-negligible number of cases. Therefore, we can assume FE to have short ciphertexts.

Then, we use a straightforward transformation to convert FE to a private-key encryption scheme SKE_q whose ciphertexts are short relative to q and whose encryption algorithm is of constant depth over $\mathbb{Z}_q$.

(2) Since the encryption algorithm of SKE_q is of constant depth, SKE_q encrypts a number x by sampling some polynomials, evaluating those polynomials at x and reducing the result modulo q. To analyse the ciphertexts of SKE_q, we need to get rid of the arithmetic overflows in the online part of its encryption algorithm. We observe that, if $r(X)$ is a polynomial with small coefficients, then, for some small x values, $r(x)$ does not change when we reduce it modulo q. Furthermore, we know the ciphertexts of SKE_q to be short relative to q. By using this fact, we can apply simple changes to the encryption algorithm of SKE_q to ensure that the polynomials sampled by its offline algorithm have very small coefficients. By doing so, we can change SKE_q to a private-key encryption scheme SKE of constant depth whose ciphertext vectors are sufficiently short and where no arithmetic overflows do occur in the online part of its encryption algorithm.

(3) In SKE, a message x gets encrypted by sampling random integer polynomials $r_1, \ldots, r_m$ of constant degree and computing $(r_1(x), \ldots, r_m(x))$ as ciphertext without any arithmetic overflows. Intuitively, such a scheme should not be secure and, indeed, we show that such a scheme can only be secure, if its ciphertexts do not contain any information about the encrypted messages. But this makes decryption impossible. Since we showed that a correct and secure FHFE scheme FE can be transformed into a secure private-key encryption scheme whose ciphertexts contain a non-negligible amount of information, it follows that FE could not be secure and correct in the first place.

We now take a closer look at the techniques used in each step.

1.4.1 Replacing Ciphertexts with Decryption Noise.

We describe here how to make the ciphertexts of FE short. For simplicity, let us assume that we have already relinearized ciphertexts and secret keys, i.e. decryption works by

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = \left\lceil \frac{\langle \mathsf{sk} \mid \mathsf{ct} \rangle}{\lfloor q/p \rfloor} \right\rfloor .$$

Query a lot of secret keys $v_1, \ldots, v_m \leftarrow \mathsf{KeyGen}(\mathsf{msk}, 0)$ for the zero-function and draw a ciphertext ct_x for an arbitrary input $x \in \mathbb{Z}_p^n$. Each v_i must decrypt ct_x to zero, since this is the value of the zero-function applied to x. Because of decryption correctness of FE, we can therefore assume that we have for each v_i

$$|\langle v_i \mid \mathsf{ct}_x \rangle| \leq \left\lfloor \frac{q}{p} \right\rfloor .$$

Otherwise, $\langle v_i \mid \mathsf{ct}_x \rangle / \lfloor q/p \rfloor$ would not round to zero. We can now exchange ct_x with the following new ciphertext for x:

$$\mathsf{ct}'_x = (\langle v_1 \mid \mathsf{ct}_x \rangle, \ldots, \langle v_m \mid \mathsf{ct}_x \rangle).$$

This ciphertext just consists of noise values which are generated when decrypting ct_x with secret keys for the zero-function. Therefore, each entry of ct'_x is bounded by $\lfloor q/p \rfloor$. The question remains, how much information about x is left in ct'_x and if it is even possible to recover $f(x)$ from ct'_x and sk_f. We show that in a non-negligible number of cases a successful decryption is still possible. That is because of the function-hiding property of FE which vaguely implies that a secret key for f has to lie in $\mathsf{span}_{\mathbb{Z}_q}\{v_1, \ldots, v_m\}$ with non-negligible probability.

1.4.2 Getting Rid of Arithmetic Overflows.

The key observation in step (2) is that, if we evaluate a polynomial of degree d with small coefficients at a small input, reducing the result modulo q will not change its value. However, the polynomials $r_1(X), \ldots, r_m(X)$ sampled in the offline part of the encryption algorithm of SKE_q do not necessarily have small coefficients. We only know them to have small output values. We prove that there is a constant c, s.t. each $c \cdot r_i$ has sufficiently small coefficients modulo q. The existence of c can be shown by using a *quasi-inverse*[1] of the Vandermonde matrix V for the tuple $(0, 1, \ldots, d)$, that is an integer matrix whose product with V equals a scaled identity matrix.

By simply multiplying ciphertexts of SKE_q with c, we can make them behave like they were outputted from an encryption algorithm of constant depth where no arithmetic overflows do occur in its online part. Therefore, we can transform SKE_q into SKE.

Quasi-inverses of Vandermonde have been recently used by Esgin et al. to extract witnesses out of many polynomial relations [ESLL19]. However, in this work, we use a different quasi-inverse than them, which yields better bounds for our results.

1.4.3 Statistically Distinguishing Random Polynomials.

We describe here, how our adversary breaks SKE in step (3). It suffices to look at the j-th coordinate of a ciphertext of SKE. At input x, the j-th coordinate is computed by sampling a random polynomial $r_j(X)$ of constant degree d in the offline part and evaluating it at x. Our adversary works by guessing one $x \neq 0$ and comparing $\mathbb{E}[r_j(x)^2]$ and $\mathbb{E}[r_j(0)^2]$. We show, if for each x the means $\mathbb{E}[r_j(x)^2]$ and $\mathbb{E}[r_j(0)^2]$ do not differ by a non-negligible amount, then $r_j(X)$ is of degree at most $d-1$ with overwhelming probability. By inductively using hybrids, one can see that $r_j(X)$ must be of degree 0, i.e. constant, with overwhelming probability. But, if $r_j(X)$ is constant, the value $r_j(x)$ does not carry any information about x. Therefore, if the ciphertexts of SKE contain a non-negligible amount of information about the encrypted messages, it follows that there must be some j and $x \neq 0$ s.t. our adversary can successfully distinguish $\mathbb{E}[r_j(x)^2]$ and $\mathbb{E}[r_j(0)^2]$ and, therefore, successfully distinguish ciphertexts for 0 from ciphertexts for x.

1.5 Organization of this Work

We first introduce some preliminaries in section 2 and some important definitions and concepts in section 3. Then, in section 4, we give an adversary who breaks private-key encryption schemes of constant depth which do not make use of arithmetic overflows. In section 5, we then derive an impossibility result for private-key encryption schemes of constant depth with short ciphertexts over $\mathbb{Z}_q$ by transforming them to schemes we broke in the preceding section. Finally, in section 6, we show the impossibility of LWE-like FHFE schemes with simple online/offline encryption by transforming them to schemes of the preceding section.

Acknowledgements. I would like to thank my doctoral supervisor Dennis Hofheinz and my former colleagues Geoffroy Couteau, Valerie Fetzer, Michael Klooß and Sven Maier for helpful comments and advices on how to improve this text. Further, I would like to thank the reviewers and everyone who listened to the talk preceding this work for their questions and suggestions.

[1] Calling such matrices quasi-inverses is ambiguous. However, we will stick to this notion, since we lack better names.

2 Preliminaries

For $n \in \mathbb{N} = \{1, 2, 3, \ldots\}$, set $[n] := \{1, \ldots, n\}$. We define two sets of functions:

$$\begin{aligned}\mathsf{poly}(\lambda) &:= \{p : \mathbb{N} \to \mathbb{N} \mid \exists c, d \in \mathbb{N}\ \forall \lambda \in \mathbb{N} :\ \lambda^c + d \geq p(\lambda) \geq 1\}\,,\\ \mathsf{negl}(\lambda) &:= \{\varepsilon : \mathbb{N} \to \mathbb{R} \mid \forall c \in \mathbb{N} :\ \lim_{\lambda\to\infty} \lambda^c \varepsilon(\lambda) = 0\}\,.\end{aligned}$$

For functions $f, g : \mathbb{N} \to \mathbb{R}$, we write

$$f(\lambda) \geq g(\lambda) - \mathsf{negl}(\lambda),$$

if there is an $\varepsilon \in \mathsf{negl}(\lambda)$ s.t. we have for all λ

$$f(\lambda) \geq g(\lambda) - \varepsilon(\lambda).$$

For $x \in \mathbb{R}$, we define the following roundings:

$$\begin{aligned}\lfloor x \rfloor &:= \max\{z \in \mathbb{Z} \mid z \leq x\}\,,\\ \lceil x \rceil &:= \min\{z \in \mathbb{Z} \mid z \geq x\}\,,\\ \lceil x \rfloor &:= \max\left\{z \in \mathbb{Z} \mid \quad |x - z| \leq \frac{1}{2}\right\}.\end{aligned}$$

For two discrete distributions $\mathcal{D}_1, \mathcal{D}_2$ over a set X we define the **statistical distance** of $(\mathcal{D}_1, \mathcal{D}_2)$ by

$$\Delta(\mathcal{D}_1, \mathcal{D}_2) := \frac{1}{2}\sum_{x\in X} |\mathcal{D}_1(x) - \mathcal{D}_2(x)|\,.$$

2.1 Statistical Preliminaries

We take the following inequality from [Hoe63].

Theorem 2 (Hoeffding's Inequality)**.** *Let $n \in \mathbb{N}$ and $B, t \geq 0$. For n independent random variables $X_1, \ldots, X_n$ with $|X_i| \leq B$, we have*

$$\Pr\left[\ \left|\frac{X_1 + \ldots + X_n}{n} - \mathbb{E}\left[\frac{X_1 + \ldots + X_n}{n}\right]\right| \geq 2Bt\ \right] \leq 2e^{-2nt^2}.$$

In this work, we will consider adversaries who approximate the means of specific random variables. By using Hoeffding's inequality, it follows that by querying a polynomial number of samples our adversaries can – with overwhelming probability – approximate the mean of a polynomially bounded random variable up to a fraction.

Corollary 2. *Let $\mathcal{D}$ be a memoryless source that outputs real numbers which are bounded by $B \geq 0$. Let $r \in \mathbb{N}$ and set $n = 2r^3$. Let μ be the mean of $\mathcal{D}$ and let E_n be the random variable which is sampled by n-fold querying $\mathcal{D}$, summing its outputs and dividing this sum by n. Then, we have*

$$\Pr\left[|E_n - \mu| \leq \frac{B}{r}\right] \geq 1 - 2e^{-r}.$$

2.2 Algebraic Preliminaries

2.2.1 Discrete Derivatives

Let $N \in \mathbb{N}$ and let $f : \{0, \ldots, N\} \to \mathbb{R}$ be a mapping.

The aim of this subsubsection is to prove the followng theorem.

Theorem 3. *Let $f(X) = \sum_{i=0}^{d} a_i X^i$ be a polynomial of degree d over $\mathbb{R}$. Then*

$$d! \cdot a_d = \sum_{k=0}^{d} (-1)^{d-k} \binom{d}{k} f(k).$$

We will prove theorem 3 by using discrete derivatives. Our strategy will work similarly to trick 2 of [GKP94], section 5.3.

Definition 1. We define the **forward difference** of f by

$$\begin{aligned} D(f) : \{0, \ldots, N-1\} &\longrightarrow \mathbb{R} \\ x &\longmapsto f(x+1) - f(x). \end{aligned}$$

Note, that the mapping $D : \mathbb{R}^{\{0,\ldots,N\}} \to \mathbb{R}^{\{0,\ldots,N-1\}}$ is $\mathbb{R}$-linear.

An easy calculation shows that the forward difference has its own Leibniz rule:

Lemma 1. *Let $x \in \{0, \ldots, N-1\}$ and $f, g : \{0, \ldots, N\} \to \mathbb{R}$. Then, we have*

$$D(f \cdot g)(x) = f(x) \cdot D(g)(x) + D(f)(x) \cdot g(x) + D(f)(x) \cdot D(g)(x).$$

More importantly, the forward difference behaves on polynomials similarly to the normal derivative:

Lemma 2. *Let $d \in \mathbb{N}$. We have:*

(a) The function $D(f)$ is zero on $\{0, \ldots, N-1\}$ iff f is constant on $\{0, \ldots, N\}$.

(b) If $f(X) = \sum_{i=0}^{d} a_i X^i$ is a polynomial with $a_0, \ldots, a_d \in \mathbb{R}$, then so is $D(f)$ and we have

$$\deg(D(f)) = \max\{0, \deg(f) - 1\}.$$

(c) If $f(X) = X^d$ and $d \leq N$, we have

$$D^d(f)(X) = d!$$

where D^d denotes the d-fold successive execution of D.

Proof. (a) $D(f)(x)$ is zero for all $x = 0, \ldots, N-1$ iff

$$f(0) = f(1) = \ldots = f(N-1) = f(N).$$

(b) First, we note for $f(X) = X^i$

$$D(f) = (X+1)^i - X^i = \sum_{k=0}^{i} \binom{i}{k} X^k - X^i = \sum_{k=0}^{i-1} \binom{i}{k} X^k.$$

Now, the claim follows by the linearity of D.

(c) We prove this claim by induction over $d \geq 1$. The base case $d = 1$ obviously holds. Now, let $d > 1$ be arbitrary with $d \leq N$. We have

$$D(f) = (X+1)^d - X^d = \sum_{k=0}^{d-1} \binom{d}{k} X^k = dX^{d-1} + \sum_{k=0}^{d-2} \binom{d}{k} X^k.$$

By the induction thesis, we have

$$D^{d-1}(X^{d-1}) = (d-1)!$$

while the $(d-1)$-th forward difference of $\sum_{k=0}^{d-2} \binom{d}{k} X^k$ vanishes according to both above claims. □

The following theorem subsumes theorem 3.

Theorem 4. *Let $d \in \{0, \dots, N\}$ and $x \in \{0, \dots, N-d\}$. For an arbitrary function $f : \{0, \dots, N\} \to \mathbb{R}$ we have*

$$D^d(f)(x) = \sum_{k=0}^{d} (-1)^{d-k} \binom{d}{k} f(x+k).$$

In particular, if $f(X) = \sum_{i=0}^{d} a_i X^i$ for some $a_i \in \mathbb{R}$, we have

$$d! \cdot a_d = D^d(f)(0) = \sum_{k=0}^{d} (-1)^{d-k} \binom{d}{k} f(k).$$

Proof. We show the claim by induction over d. For $d = 0, 1$ the claim trivially holds.

Now, let $d > 1$. We then have by the induction hypothesis

$$D^{d-1}(f)(x) = \sum_{k=0}^{d-1} (-1)^{d-1-k} \binom{d-1}{k} f(x+k).$$

If we set $f_k(X) := f(X+k)$, we get by applying D again

$$\begin{aligned}
D^d(f)(x) =& D\left(\sum_{k=0}^{d-1} (-1)^{d-1-k} \binom{d-1}{k} f_k \right)(x) \\
=& \sum_{k=0}^{d-1} (-1)^{d-1-k} \binom{d-1}{k} D(f_k)(x) \\
=& \sum_{k=0}^{d-1} (-1)^{d-1-k} \binom{d-1}{k} (f_k(x+1) - f_k(x)) \\
=& \sum_{k=0}^{d-1} (-1)^{d-1-k} \binom{d-1}{k} f(x+k+1) - \sum_{k=0}^{d-1} (-1)^{d-1-k} \binom{d-1}{k} f(x+k)
\end{aligned}$$

We can rewrite the last term as

$$\sum_{k=0}^{d-1} (-1)^{d-1-k} \binom{d-1}{k} f(x+k+1) - \sum_{k=0}^{d-1} (-1)^{d-1-k} \binom{d-1}{k} f(x+k) =: \sum_{k=0}^{d} b_k f(x+k)$$

by setting

$$b_k = \begin{cases}
-(-1)^{d-1} \binom{d-1}{0}, & \text{if } k = 0, \\
(-1)^{d-1-(k-1)} \binom{d-1}{k-1} - (-1)^{d-1-k} \binom{d-1}{k}, & \text{if } d-1 \geq k \geq 1, \\
(-1)^{d-1-(d-1)} \binom{d-1}{d-1}, & \text{if } k = d.
\end{cases}$$

Easy calculations show now for all $k = 0, \dots, d$

$$b_k = (-1)^{d-k} \binom{d}{k}.$$

□

2.2.2 Modulo Valuation

Now, let $q \in \mathbb{N}$ be a modulus.

Definition 2. For $a \in \mathbb{Z}$, we define the **absolute value modulo** q by

$$|a \bmod q| := \min_{z \in q\mathbb{Z}} |a + z| \in \left\{ 0, \dots, \left\lfloor \frac{q}{2} \right\rfloor \right\}.$$

Lemma 3. *(a) For $a \in \mathbb{Z}$, we have $|a \bmod q| = 0 \Leftrightarrow a \in q\mathbb{Z}$.*

(b) For $a_1, \ldots, a_n \in \mathbb{Z}$, we have $|\sum_{i=1}^n a_i \bmod q| \leq \sum_{i=1}^n |a_i \bmod q|$.

(c) For $a, z \in \mathbb{Z}$, we have $|z \cdot a \bmod q| \leq |z| \cdot |a \bmod q|$.

Proof. Both inequalities (b) and (c) follow, if we can show for $a, b \in \mathbb{Z}$

$$|a + b \bmod q| \leq |a \bmod q| + |b \bmod q|\,.$$

We can rewrite this inequality to

$$\min_{z \in q\mathbb{Z}} |a + b + z| \leq \min_{z_1 \in q\mathbb{Z}} |a + z_1| + \min_{z_2 \in q\mathbb{Z}} |b + z_2|.$$

Let $z_1^*, z_2^* \in q\mathbb{Z}$ be optimal for the terms on the right hand. Then

$$\min_{z \in q\mathbb{Z}} |a + b + z| \leq |a + b + z_1^* + z_2^*| \leq |a + z_1^*| + |b + z_2^*| = \min_{z_1 \in q\mathbb{Z}} |a + z_1| + \min_{z_2 \in q\mathbb{Z}} |b + z_2|. \qquad \square$$

2.3 Learning Theory-Preliminaries

In this subsection, we study the problem of learning vector subspaces. Let $\mathbb{F}$ be an arbitrary field.

Lemma 4. *Let $s \in \mathbb{N}_0 = \{0, 1, 2, \ldots\}$ and let $\mathcal{D}$ be a discrete distribution over $\mathbb{F}^s$. For $m \in \mathbb{N}$, we have*

$$\Pr_{v_1, \ldots, v_m \leftarrow \mathcal{D}} \left[v_m \in \mathsf{span}_{\mathbb{F}} \{v_1, \ldots, v_{m-1}\}\right] \geq 1 - \frac{s}{m}.$$

Proof. Let $m > s$ and fix $v_1, \ldots, v_m \in \mathsf{supp}(\mathcal{D})$. Denote by S^m the group of permutations of the set $[m]$ and by $T \subset S^m$ the subgroup of order m which is generated by the cyclic rotation $(123 \ldots m)$. For $\tau \in T$ set

$$V_\tau := \mathsf{span}_{\mathbb{F}} \left\{v_{\tau(1)}, \ldots, v_{\tau(m-1)}\right\}.$$

Since each v_i is an s-dimensional vector, we have

$$m - s \leq \#\left\{j \in [m] \mid v_j \in \mathsf{span}_{\mathbb{F}} \{v_i \mid i \in [m] \setminus \{j\}\}\right\} = \#\left\{\tau \in T \mid v_{\tau(m)} \in V_\tau\right\}.$$

Therefore, for each fixed choice $v_1, \ldots, v_m \in \mathsf{supp}(\mathcal{D})$ we have

$$\Pr_{\tau \leftarrow T} \left[v_{\tau(m)} \in V_\tau\right] \geq \frac{m - s}{m}.$$

Since the vectors $v_1, \ldots, v_m$ are identically and independently distributed, we furthermore have

$$\Pr_{v_1, \ldots, v_m \leftarrow \mathcal{D}} \left[v_m \in \mathsf{span}_{\mathbb{F}} \{v_1, \ldots, v_{m-1}\}\right] = \Pr_{\substack{v_1, \ldots, v_m \leftarrow \mathcal{D} \\ \tau \leftarrow T}} \left[v_{\tau(m)} \in V_\tau\right].$$

Combining both things, we get

$$\begin{aligned}
&\Pr_{v_1, \ldots, v_m \leftarrow \mathcal{D}} \left[v_m \in \mathsf{span}_{\mathbb{F}} \{v_1, \ldots, v_{m-1}\}\right] = \Pr_{\substack{v_1, \ldots, v_m \leftarrow \mathcal{D} \\ \tau \leftarrow T}} \left[v_{\tau(m)} \in V_\tau\right] \\
&= \sum_{v_1, \ldots, v_m \in \mathsf{supp}(\mathcal{D})} \Pr_{\tau \leftarrow T} \left[v_{\tau(m)} \in V_\tau\right] \cdot \Pr_{w_1, \ldots, w_m \leftarrow \mathcal{D}} \left[\forall i : \; w_i = v_i\right] \\
&\geq \sum_{v_1, \ldots, v_m \in \mathsf{supp}(\mathcal{D})} \frac{m - s}{m} \cdot \Pr_{w_1, \ldots, w_m \leftarrow \mathcal{D}} \left[\forall i : \; w_i = v_i\right] = \frac{m - s}{m}.
\end{aligned} \qquad \square$$

Theorem 5. *Let $s \in \mathbb{N}_0$ and let $\mathcal{D}$ be a discrete distribution over $\mathbb{F}^s$. Then, there exists an algorithm which makes s queries to $\mathcal{D}$ and $O(s^3)$-fold use of the four basic arithmetic operations in $\mathbb{F}$ to compute a number $k \leq s$, a matrix $B \in \mathbb{F}^{s \times k}$ which consists of k samples of $\mathcal{D}$ and a second matrix $B^+ \in \mathbb{F}^{k \times s}$ s.t. with $V := B \cdot \mathbb{F}^k$*

(a) we have $B^+ \cdot B = 1_{k \times k}$,

(b) $B \cdot B^+$ *is the identity on* V*, i.e., for all* $v \in V$*, we have* $B \cdot B^+ \cdot v = v$*,*

(c) a certain proportion of the samples of $\mathcal{D}$ *lies in* V*, i.e.* $\Pr_{v \leftarrow \mathcal{D}}[v \in V] \geq \frac{1}{s}$*.*

Proof. Our algorithm samples first $v_1, \ldots, v_s \leftarrow \mathcal{D}$ and then chooses a basis $b_1, \ldots, b_k \in \{v_1, \ldots, v_s\}$ of $V = \mathsf{span}_{\mathbb{F}}\{v_1, \ldots, v_s\}$. This can be achieved by using the Gaussian elimination algorithm which makes use of $O(s^3)$ basic arithmetic operations of $\mathbb{F}$. Claim (c) then follows by lemma 4. The algorithm then sets

$$B := (b_1 | \ldots | b_k) \in \mathbb{F}^{s \times k}$$

and computes a matrix $B^+ \in \mathbb{F}^{k \times s}$ s.t. $B^+ \cdot B = 1_{k \times k}$. Such a matrix B^+ can be computed by choosing a subset of k linearly independent rows of B, computing an inverse to them and filling this inverse column-wise with zeros s.t. we get the desired property. For this task, our algorithm needs to perform $O(s^3)$ basic arithmetic operations in $\mathbb{F}$. Therefore, claim (a) does follow.

Claim (b) is implied by (a). □

3 Definitions

In this section, we give basic definitions and state elementary lemmas for this work.

3.1 Functional Encryption

Throughout this work, let λ denote the security parameter. Let $(F_\lambda)_\lambda$ be a family of function descriptions with a family of domains $(X_\lambda)_\lambda$ and codomains $(Y_\lambda)_\lambda$. We tacitly assume in the following that the size of each $f \in F_\lambda, x \in X_\lambda$ and $y \in Y_\lambda$ is bounded by a polynomial in λ, that we can efficiently sample uniformly random elements of those families and that there is a deterministic polytime evaluation algorithm which on input $(f, x) \in F_\lambda \times X_\lambda$ outputs the correct value $y \in Y_\lambda$. We denote the output of this algorithm by $f(x)$.

Definition 3. A **functional encryption scheme** $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ for the family $(F_\lambda)_\lambda$ is a quadruple of four ppt algorithms where

$\mathsf{Setup}(1^\lambda)$ on input 1^λ generates a master secret key msk,

$\mathsf{KeyGen}(\mathsf{msk}, f)$ on input msk and a function $f \in F_\lambda$ generates a secret key sk_f,

$\mathsf{Enc}(\mathsf{msk}, x)$ on input msk and an input value $x \in X_\lambda$ generates a ciphertext ct_x,

$\mathsf{Dec}(\mathsf{sk}_f, \mathsf{ct}_x)$ on input a secret key sk_f and a ciphertext ct_x outputs a value $y \in Y_\lambda$.

We call FE **correct**, if we have for each *samplable*[2] $(f_\lambda)_\lambda \in (F_\lambda)_\lambda$ an $\varepsilon \in \mathsf{negl}(\lambda)$, s.t. it holds for all $(x_\lambda)_\lambda \in (X_\lambda)_\lambda$

$$\Pr\left[\mathsf{Dec}\left(\mathsf{sk}_f, \mathsf{ct}_x\right) = f_\lambda(x_\lambda) \;\middle|\; \begin{array}{l} \mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda), \\ \mathsf{sk}_f \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_\lambda), \\ \mathsf{ct}_x \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda) \end{array}\right] \geq 1 - \varepsilon(\lambda).$$

We call FE **better than guessing** (by $\frac{1}{r}$), if there exists a polynomial $r \in \mathsf{poly}(\lambda)$ s.t. we have for each $(x_\lambda)_\lambda \in (X_\lambda)_\lambda$ and each samplable $(f_\lambda)_\lambda \in (F_\lambda)_\lambda$

$$\Pr\left[\mathsf{Dec}\left(\mathsf{sk}_f, \mathsf{ct}_x\right) = f_\lambda(x_\lambda) \;\middle|\; \begin{array}{l} \mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{sk}_f \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_\lambda), \\ \mathsf{ct}_x \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda) \end{array}\right] \geq \frac{1}{r(\lambda)} + \frac{1}{\#Y_\lambda} - \mathsf{negl}(\lambda).$$

We call FE **useless**, if we have for each polynomial $r \in \mathsf{poly}(\lambda)$

$$\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)}\left[\forall x, y \in X_\lambda : \; \Delta\left(\mathsf{Enc}(\mathsf{msk}, x), \mathsf{Enc}(\mathsf{msk}, y)\right) < \frac{1}{r(\lambda)}\right] \geq 1 - \mathsf{negl}(\lambda).$$

While being correct is a common requirement for encryption schemes, being useless implies that a successful decryption is almost impossible, since the ciphertexts contain nearly no information. Being better than guessing, however, implies that in some cases the ciphertexts and secret keys contain enough information for a successful decryption. Now, one would assume that a scheme cannot be useless and better than guessing at the same time and, indeed, we have the following lemma:

Lemma 5. *Let $\#Y_\lambda \geq 2$ for all λ and let $(F_\lambda)_\lambda$ contain a samplable $(f_\lambda)_\lambda$ s.t. each f_λ is surjective. Then, we have:*

(a) If FE is correct, it is better than guessing.

(b) If FE is useless, it is not better than guessing.

[2] By being **samplable**, we mean here that there is a uniform deterministic poly-time algorithm which on input 1^λ outputs f_λ.

Proof. To see (a), it suffices to take $r(\lambda) = 2$.

For (b), assume for the sake of contradiction that FE is both useless and better than guessing. Let $(f_\lambda)_\lambda$ s.t. each f_λ is surjective. Then, for $\mathsf{msk} \in \mathsf{supp}(\mathsf{Setup}(1^\lambda))$ and $x \in X_\lambda$, define the distribution

$$H_{\mathsf{msk}}(x) := \mathsf{Dec}(\mathsf{KeyGen}(\mathsf{msk}, f_\lambda), \mathsf{Enc}(\mathsf{msk}, x)).$$

For $a, b \in X_\lambda$, set

$$p_{a,b} := \Pr_{\mathsf{msk} \leftarrow \mathcal{D}} \left[H_{\mathsf{msk}}(a) = f_\lambda(b) \right].$$

Since FE is better than guessing, there is an $r \in \mathsf{poly}(\lambda)$ s.t. for each $(x_\lambda)_\lambda$ we have

$$p_{x_\lambda, x_\lambda} = \Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)} \left[H_{\mathsf{msk}}(x_\lambda) = f_\lambda(x_\lambda) \right] \geq \frac{1}{\#Y_\lambda} + \frac{1}{r(\lambda)} - \mathsf{negl}(\lambda). \tag{4}$$

Since FE is useless, we have

$$\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)} \left[\forall x, y \in X_\lambda : \ 2\Delta \left(H_{\mathsf{msk}}(x), H_{\mathsf{msk}}(y) \right) < \frac{1}{4r(\lambda)} \right] \geq 1 - \mathsf{negl}(\lambda).$$

Let $\mathcal{D}$ be the output of $\mathsf{Setup}(1^\lambda)$ conditioned on

$$\forall x, y \in X_\lambda : \ 2\Delta \left(H_{\mathsf{msk}}(x), H_{\mathsf{msk}}(y) \right) < \frac{1}{4r(\lambda)}. \tag{5}$$

If λ is big enough, we have for each $x \in X_\lambda$

$$p_{x,x} := \Pr_{\mathsf{msk} \leftarrow \mathcal{D}} \left[H_{\mathsf{msk}}(x) = f_\lambda(x) \right] \geq \frac{1}{\#Y_\lambda} + \frac{1}{2r(\lambda)}.$$

Fix an $a \in X_\lambda$. Because of inequality (4) and since $f_\lambda : X_\lambda \to Y_\lambda$ is surjective, there must exist a $b \in X_\lambda$ s.t.

$$p_{a,b} := \Pr_{\mathsf{msk} \leftarrow \mathcal{D}} \left[H_{\mathsf{msk}}(a) = f_\lambda(b) \right] < \frac{1}{\#Y_\lambda}.$$

Since $\mathcal{D}$ is conditioned on (5), we have $|p_{a,c} - p_{b,c}| \leq 2\Delta \left(H_{\mathsf{msk}}(a), H_{\mathsf{msk}}(b) \right) < \frac{1}{4r}$ for all a, b, c. In particular, we get the contradiction

$$\frac{1}{\#Y_\lambda} + \frac{1}{2r(\lambda)} \leq p_{b,b} \leq p_{a,b} + \frac{1}{4r(\lambda)} < \frac{1}{\#Y_\lambda} + \frac{1}{4r(\lambda)}.$$

□

3.2 Encryption Algorithms

Now, let R be a ring with an associated valuation $|\cdot|_R : R \to \mathbb{N}_0$. In this work, we always assume $R = \mathbb{Z}$ or $R = \mathbb{Z}_q$ for a prime $q = q(\lambda)$. In the first case $|\cdot|_{\mathbb{Z}} = |\cdot|$ is the archimedean absolute value. In the latter case $|\cdot|_{\mathbb{Z}_q} = |\cdot \bmod q|$ is the absolute value modulo q we defined in definition 2.

Furthermore, let $X_\lambda = \{0, \ldots, N\}^n$ now consist of n-dimensional vectors for a polynomial $n = n(\lambda) \in \mathsf{poly}(\lambda)$ and some $N = N(\lambda)$.

Definition 4. We say the scheme FE or rather its encryption algorithm Enc is of **length** s over R, if the output of Enc is always an element of R^s. Furthermore, we say in this case that Enc is of

(a) **width** B, if the infinity-norm of almost all ciphertexts is bounded by B. I.e., there is an $\varepsilon \in \mathsf{negl}(\lambda)$, s.t. we have for each $(x_\lambda)_\lambda \in (X_\lambda)_\lambda$

$$\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)} \left[\ \exists i \in [s] : |c_i|_R > B \mid c \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda) \right] \leq \varepsilon(\lambda),$$

(b) **depth** d, if Enc consists of two parts: an **offline part** – a ppt algorithm $\mathsf{Enc}_{\text{offline}}$ which on input msk generates s polynomials over $R[X_1, \ldots, X_n]$ of total degree $\leq d$ – and an **online**

part which generates a ciphertext by evaluating the polynomials sampled by $\mathsf{Enc}_{\mathsf{offline}}$ at the input x. I.e., Enc works as follows

$$\begin{array}{l}\underline{\mathsf{Enc}(\mathsf{msk}, x):} \\ (p_1, \ldots, p_s) \leftarrow \mathsf{Enc}_{\mathsf{offline}}(\mathsf{msk}) \\ \mathsf{ct}_x := (p_1(x), \ldots, p_s(x)) \\ \underline{\mathsf{return}\ \mathsf{ct}_x}\end{array}$$

where we demand that each p_i is a polynomial of total degree $\leq d$ over R.

3.3 Security Notions

In this work, we study the notion of *selective* and *function-hiding* IND-CPA security where the adversary is allowed to submit a priori multiple challenge inputs (x_i^0, x_i^1) and a bounded number of challenge functions (f_j^0, f_j^1). To be feasible, the adversary must ensure that the output values $f_j^b(x_i^b)$ do not already tell him, if he lives in world 0 or world 1, i.e. he must ensure $f_j^0(x_i^0) = f_j^1(x_i^1)$. The challenger will send the adversary the ciphertexts and secret keys for one random bit $b \leftarrow \{0,1\}$. To win, the adversary has to guess the bit b.

Definition 5. Let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be a functional encryption scheme for the family $(F_\lambda)_\lambda$ and let $m \in \mathsf{poly}(\lambda)$. We say that FE is **selectively m-bounded function-hiding IND-CPA secure** (**m-fh-IND-CPA secure**), if each ppt adversary $\mathcal{A}$ has a negligible advantage in winning the following game:

Step 1: The adversary $\mathcal{A}$ submits two lists[3] of possible inputs $(x_i^0)_{i=1}^n, (x_i^1)_{i=1}^n$ and two lists of possible functions $(f_j^0)_{j=1}^m, (f_j^1)_{j=1}^m$ to the challenger $\mathcal{C}$.

Step 2: The challenger $\mathcal{C}$ generates a master secret key $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)$ and draws a secret bit $b \leftarrow \{0,1\}$. Then, $\mathcal{C}$ computes $\mathsf{ct}_{x_i^b} := \mathsf{Enc}(\mathsf{msk}, x_i^b)$ for each $i = 1, \ldots, n$, $\mathsf{sk}_{f_j^b} := \mathsf{KeyGen}(\mathsf{msk}, f_j^b)$ for each $j = 1, \ldots, m$ and sends the lists $(\mathsf{ct}_{x_i^b})_{i=1}^n$ and $(\mathsf{sk}_{f_j^b})_{j=1}^m$ to $\mathcal{A}$.

Step 3: The adversary $\mathcal{A}$ guesses b.

The adversary wins the above game, if he guesses b correctly, and, if we have $f_j^0(x_i^0) = f_j^1(x_i^1)$ for all $i = 1, \ldots, n$ and $j = 1, \ldots, m$. The **advantage** of $\mathcal{A}$ is defined by

$$\mathsf{Adv}(\mathcal{A}) := 2\Pr[\mathcal{A} \text{ wins}] - 1 = \Pr[\mathcal{A} \text{ wins} \mid b = 0] + \Pr[\mathcal{A} \text{ wins} \mid b = 1] - 1.$$

We call FE **selectively unbounded function-hiding IND-CPA secure** (**fh-IND-CPA secure**), if FE is m-fh-IND-CPA secure for each polynomial $m \in \mathsf{poly}(\lambda)$, and we call FE **selectively IND-CPA secure** (**IND-CPA secure**), if FE is 0-fh-IND-CPA secure.

3.4 Private-Key Encryption

We define private-key encryption schemes as a special case of functional encryption schemes:

Definition 6. A **private-key encryption scheme** is a functional encryption scheme $\mathsf{SKE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ for a function family $(F_\lambda)_\lambda$ where each F_λ only contains the identity function $\mathrm{Id} : X_\lambda \to X_\lambda$.

When discussing private-key encryption schemes we sometimes omit KeyGen from the header of the scheme and write $\mathsf{Dec}(\mathsf{msk}, \cdot)$ instead of $\mathsf{Dec}(\mathsf{KeyGen}(\mathsf{msk}, \mathrm{Id}), \cdot)$. Note that we call SKE IND-CPA secure, if it is selectively 0-bounded function-hiding IND-CPA secure in the sense of definition 5. This differs from the usual security notion in literature, where the adversary is usually allowed to submit only one pair of challenge messages and can inquire ciphertexts adaptively. However, by using a hybrid argument, one can show that the security loss which occurs by allowing multiple challenge messages is polynomially bounded. If we consider message spaces of superpoly size, then we can construct private-key encryption schemes which are selectively, but not adaptively, secure. Therefore, the security notion for SKE we use here is weaker than the usual one in literature.

[3]The size n is determined by the description of $\mathcal{A}$ and bounded by $\mathcal{A}$'s running time. The size n may be zero, which would mean that $\mathcal{A}$ always sends two empty lists of inputs.

3.5 Transformations

Definition 7. Let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$, $\mathsf{FE}' = (\mathsf{Setup}', \mathsf{KeyGen}', \mathsf{Enc}', \mathsf{Dec}')$ be two functional encryption schemes for the same functionality. We say that FE is **virtually** FE', if the following algorithms are equal

$$\begin{aligned}\mathsf{Setup} &= \mathsf{Setup}'\\ \mathsf{KeyGen} &= \mathsf{KeyGen}'\\ \mathsf{Dec} &= \mathsf{Dec}'\end{aligned}$$

and if there is an $\varepsilon \in \mathsf{negl}(\lambda)$, s.t. for all sequences $(x_\lambda)_\lambda \in (X_\lambda)_\lambda$ the statistical distance between the following two distributions is bounded from above by ε:

$$\begin{aligned}&\left\{(\mathsf{msk}, \mathsf{ct}_x) \mid \mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda), \mathsf{ct}_x \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda)\right\},\\ &\left\{(\mathsf{msk}, \mathsf{ct}'_x) \mid \mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda), \mathsf{ct}'_x \leftarrow \mathsf{Enc}'(\mathsf{msk}, x_\lambda)\right\}.\end{aligned}$$

Now, let FE be a functional encryption scheme for functions (F_λ) with inputs (X_λ) and let FE' be one for functions (F'_λ) with inputs (X'_λ). We say there is an **adversarial transformation** from FE to FE', if there are ppt algorithms $\mathcal{T}_{\mathsf{ct}}, \mathcal{T}_{\mathsf{sk}}, \mathcal{T}_F, \mathcal{T}_X$ s.t. we have the following equalities of distributions for all $x' \in X'_\lambda$, $f' \in F'_\lambda$, $\mathsf{msk} \in \mathsf{supp}(\mathsf{Setup})$:

$$\begin{aligned}\mathsf{Setup}'(1^\lambda) &= \mathsf{Setup}(1^\lambda),\\ \mathsf{Enc}'(\mathsf{msk}, x') &= \mathcal{T}_{\mathsf{ct}}(\mathsf{Enc}(\mathsf{msk}, \mathcal{T}_X(x'))),\\ \mathsf{KeyGen}'(\mathsf{msk}, f') &= \mathcal{T}_{\mathsf{sk}}(\mathsf{KeyGen}(\mathsf{msk}, \mathcal{T}_F(f'))).\end{aligned}$$

If $(F_\lambda) = (F'_\lambda)$, then we always assume $\mathcal{T}_F = \mathrm{Id}_{F_\lambda}$ and $\mathcal{T}_X = \mathrm{Id}_{X_\lambda}$.

Let $k \in \mathbb{N}$ be constant and let $(\mathsf{FE}^i)_{i=1}^k$ be a sequence of functional encryption schemes. We say there is a **virtual adversarial transformation** from FE^1 to FE^k, if, for each $i = 1, \ldots, k-1$, FE^i is virtually FE^{i+1} or there is an adversarial transformation from FE^i to FE^{i+1}.

We can now observe the following facts:

Lemma 6. *(a) If FE is virtually FE', then FE is m-fh-IND-CPA secure, correct, better than guessing resp. useless iff FE' is so.*

(b) If FE is m-fh-IND-CPA secure and there is an adversarial transformation from FE to FE', then FE' is m-fh-IND-CPA secure.

Proof. (a) Let FE be virtually FE'. Then, it is clear that, if FE is correct, then FE' is correct, too. The same goes for being better than guessing and being useless.

Now, let FE be m-fh-IND-CPA secure. We have to show that FE' is m-fh-IND-CPA secure, too. Let $\mathcal{A}$ be a selective adversary who plays the IND-CPA game from definition 5 with FE'. We have to show that $\mathcal{A}$ does not notice when we exchange FE' with FE. W.l.o.g., we can assume that there is a polynomial $s \in \mathsf{poly}(\lambda)$ s.t. $\mathcal{A}$ requires ciphertexts for s messages $x_1, \ldots, x_s \in X_\lambda$ and secret keys for m functions $f_1, \ldots, f_m \in F_\lambda$.

Since FE is virtually FE', there is an $\varepsilon(\lambda) \in \mathsf{negl}(\lambda)$, s.t. for all $(x_\lambda)_\lambda$, for $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda), \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda), \mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}, x_\lambda)$ we have

$$\begin{aligned}&2\Delta((\mathsf{msk}, \mathsf{ct}), (\mathsf{msk}, \mathsf{ct}'))\\ &= \sum_{\mathsf{msk}^*}\sum_{\mathsf{ct}^*}\left|\Pr\left[\mathsf{msk}^* = \mathsf{Setup}(1^\lambda), \mathsf{ct}^* = \mathsf{Enc}(\mathsf{msk}^*, x_\lambda)\right] - \Pr\left[\mathsf{msk}^* = \mathsf{Setup}(1^\lambda), \mathsf{ct}^* = \mathsf{Enc}'(\mathsf{msk}^*, x_\lambda)\right]\right|\\ &= \sum_{\mathsf{msk}^*}\Pr\left[\mathsf{msk}^* = \mathsf{Setup}(1^\lambda)\right] \cdot \sum_{\mathsf{ct}^*}\left|\Pr\left[\mathsf{ct}^* = \mathsf{Enc}(\mathsf{msk}^*, x_\lambda)\right] - \Pr\left[\mathsf{ct}^* = \mathsf{Enc}'(\mathsf{msk}^*, x_\lambda)\right]\right| \leq \varepsilon(\lambda).\end{aligned}$$

For $k \in \{0, 1, \ldots, s\}$, draw $\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda), \mathsf{sk}_1 \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_1), \ldots, \mathsf{sk}_m \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f_m)$ and

$$\mathsf{ct}_i \leftarrow \begin{cases}\mathsf{Enc}'(\mathsf{msk}, x_i), & \text{if } i \leq k,\\ \mathsf{Enc}(\mathsf{msk}, x_i), & \text{if } i > k.\end{cases}$$

If we set $V_k := (\mathsf{msk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_m, \mathsf{ct}_1, \ldots, \mathsf{ct}_s)$, then V_0 equals the view of $\mathcal{A}$ in the selective IND-CPA game against FE, while V_s equals the view of $\mathcal{A}$ in the selective IND-CPA game against FE'. Now, fix

$$\begin{aligned} & \mathsf{msk}^* \in \mathsf{supp}(\mathsf{Setup}(1^\lambda)), \\ \text{for } i = 1, \ldots, s, \quad & \mathsf{ct}_i^* \in \mathsf{supp}(\mathsf{Enc}(\mathsf{msk}^*, x_i)) \cup \mathsf{supp}(\mathsf{Enc}'(\mathsf{msk}^*, x_i)), \\ \text{for } j = 1, \ldots, m, \quad & \mathsf{sk}_j^* \in \mathsf{supp}(\mathsf{KeyGen}(\mathsf{msk}^*, f_j)). \end{aligned}$$

Then, we have

$$\begin{aligned} & \Pr\left[V_k = (\mathsf{msk}^*, \mathsf{sk}_1^*, \ldots, \mathsf{sk}_m^*, \mathsf{ct}_1^*, \ldots, \mathsf{ct}_s^*)\right] \\ = & \Pr\left[\mathsf{msk}^* = \mathsf{Setup}(1^\lambda)\right] \cdot \prod_{j=1}^{m} \Pr\left[\mathsf{sk}_j^* = \mathsf{KeyGen}(\mathsf{msk}^*, f_j)\right] \\ & \cdot \prod_{i=1}^{k} \Pr\left[\mathsf{ct}_i^* = \mathsf{Enc}'(\mathsf{msk}^*, x_i)\right] \cdot \prod_{i=k+1}^{s} \Pr\left[\mathsf{ct}_i^* = \mathsf{Enc}(\mathsf{msk}^*, x_i)\right]. \end{aligned}$$

For $k < s$, it follows now

$$\begin{aligned} & 2\Delta(V_k, V_{k+1}) \\ = & \sum_{\mathsf{msk}^*} \Pr\left[\mathsf{msk}^* = \mathsf{Setup}(1^\lambda)\right] \prod_{j=1}^{m} \left(\sum_{\mathsf{sk}_j^*} \Pr\left[\mathsf{sk}_j^* = \mathsf{KeyGen}(\mathsf{msk}^*, f_j)\right] \right) \\ & \cdot \prod_{i=1}^{k} \left(\sum_{\mathsf{ct}_i^*} \Pr\left[\mathsf{ct}_i^* = \mathsf{Enc}'(\mathsf{msk}^*, x_i)\right] \right) \cdot \prod_{i=k+2}^{s} \left(\sum_{\mathsf{ct}_i^*} \Pr\left[\mathsf{ct}_i^* = \mathsf{Enc}(\mathsf{msk}^*, x_i)\right] \right) \\ & \cdot \left| \Pr\left[\mathsf{ct}_{k+1}^* = \mathsf{Enc}'(\mathsf{msk}^*, x_{k+1})\right] - \Pr\left[\mathsf{ct}_{k+1}^* = \mathsf{Enc}(\mathsf{msk}^*, x_{k+1})\right] \right| \\ = & \sum_{\mathsf{msk}^*} \Pr\left[\mathsf{msk}^* = \mathsf{Setup}(1^\lambda)\right] \cdot \left| \Pr\left[\mathsf{ct}_{k+1}^* = \mathsf{Enc}'(\mathsf{msk}^*, x_{k+1})\right] - \Pr\left[\mathsf{ct}_{k+1}^* = \mathsf{Enc}(\mathsf{msk}^*, x_{k+1})\right] \right| \leq \varepsilon(\lambda). \end{aligned}$$

Using triangle inequality, we get

$$2\Delta(V_0, V_s) \leq \sum_{k=0}^{s-1} \Delta(V_k, V_{k+1}) \leq s \cdot \varepsilon(\lambda) \in \mathsf{negl}(\lambda).$$

Therefore, the advantage of $\mathcal{A}$ to distinguish the IND-CPA game with FE from the IND-CPA game with FE' is statistically bounded by a negligible value.

Therefore, FE' is m-fh-IND-CPA secure.

(b) Let $\mathcal{A}'$ be a selective adversary who plays the IND-CPA game from definition 5 with FE'. We have to construct a selective adversary $\mathcal{A}$ who plays the same IND-CPA game with FE and whose advantage against FE is at least as big as the advantage of $\mathcal{A}'$ against FE'.

The adversary $\mathcal{A}$ proceeds as follows:

Step 1: He starts $\mathcal{A}'$ as a subroutine and collects all ciphertext-queries ${x'_1}^0, \ldots, {x'_s}^0, {x'_1}^1, \ldots, {x'_s}^1$ and all secret key-queries ${f'_1}^0, \ldots, {f'_m}^0, {f'_1}^1, \ldots, {f'_m}^1$ of $\mathcal{A}'$. Then, he computes for $b = 0, 1$ and $i = 1, \ldots, s$

$$x_i^b := \mathcal{T}_X({x'_i}^b)$$

and for $j = 1, \ldots, m$

$$f_j^b := \mathcal{T}_F({f'_j}^b).$$

Then, he submits the inputs ${x_1}^0, \ldots, {x_s}^0, {x_1}^1, \ldots, {x_s}^1$ and the functions ${f_1}^0, \ldots, {f_m}^0$, ${f_1}^1, \ldots, {f_m}^1$ to the challenger.

Step 2: The challenger draws $b^* \leftarrow \{0,1\}$ and sends to $\mathcal{A}$ the ciphertexts

$$\mathsf{ct}_i = \mathsf{Enc}(\mathsf{msk}, x_i^{b^*}) = \mathsf{Enc}(\mathsf{msk}, \mathcal{T}_X(x_i'^{b^*}))$$

and the secret keys

$$\mathsf{sk}_j = \mathsf{KeyGen}(\mathsf{msk}, f_j^{b^*}) = \mathsf{KeyGen}(\mathsf{msk}, \mathcal{T}_F(f_j'^{b^*})).$$

$\mathcal{A}$ computes now for $i = 1, \ldots, s$

$$\mathsf{ct}_i' := \mathcal{T}_{\mathsf{ct}}(\mathsf{ct}_i)$$

and for $j = 1, \ldots, m$

$$\mathsf{sk}_j' := \mathcal{T}_{\mathsf{sk}}(\mathsf{sk}_j)$$

and sends the ciphertext $\mathsf{ct}_1', \ldots, \mathsf{ct}_s'$ and the secret keys $\mathsf{sk}_1', \ldots, \mathsf{sk}_m'$ to $\mathcal{A}'$.

Step 3: $\mathcal{A}$ outputs the bit which is guessed by $\mathcal{A}'$.

Since $\mathcal{T}_{\mathsf{ct}}, \mathcal{T}_{\mathsf{sk}}, \mathcal{T}_F, \mathcal{T}_X$ form an adversarial transformation, the view of $\mathcal{A}'$ while he is being used by $\mathcal{A}$ in the IND-CPA game against FE is identically distributed as the view of $\mathcal{A}'$ while he plays the IND-CPA game against FE'. Therefore, the advantage of $\mathcal{A}$ against FE equals the advantage of $\mathcal{A}'$ against FE'. □

At some points, we want to ensure that an encryption algorithm Enc of width B never outputs a ciphertext whose largest entry is not bounded by B. We can ensure such a behaviour by replacing each ciphertext of Enc which is too big with the zero vector. It is clear that this change just has a statistically negligible impact on a scheme. One can even ensure that by doing so we do not harm the depth of Enc:

Lemma 7. *For $n = 1$, let FE be of length s, width B and depth d over R. If d is constant and B is polynomial, then FE is virtually a scheme $\mathsf{FE}' = (\mathsf{Setup}', \mathsf{KeyGen}', \mathsf{Enc}', \mathsf{Dec}')$ of length s and depth d over R where we have $\mathsf{Enc}'(\mathsf{msk}', x) \in \{-B, \ldots, B\}^s$ for all λ, $x \in X_\lambda$ and $\mathsf{msk}' \in \mathsf{supp}(\mathsf{Setup}'(1^\lambda))$.*

Before we can prove lemma 7, we need the following algebraic lemma which states that one can ensure that a polynomial of constant degree is bounded by just observing its behaviour on a set of polynomial size.

Lemma 8. *Let $p \in R[X]$ be a polynomial of degree d and let $N, B \in \mathbb{N}$. If we set*

$$I = \{0, \ldots, \min(N, 2(d+1)(2B+1)^{d+1} - 1)\},$$

then we have

$$\exists x \in \{0, \ldots, N\}^n : \ |p(x)|_R > B \iff \exists x \in I : \ |p(x)|_R > B.$$

Proof. Since $I \subset \{0, \ldots, N\}^n$, the direction from right to left does hold.

To prove the other direction assume, for the sake of contradicition, there is an $x^* \in \{0, \ldots, N\}^n$ with $|p(x^*)|_R > B$, but for all $x \in I$ we have $|p(x)|_R \leq B$.

Since p is a polynomial of degree d over $\mathbb{Z}$ or $\mathbb{Z}_q$ for some prime q, we have that for each x the value $p(x + d + 1)$ is determined by the values

$$p(x), \ldots, p(x + d).$$

Therefore, if there are integers $x \neq y$ with $p(x+i) = p(y+i)$ for all $i = 0, \ldots, d$, then p is periodic with a period $\leq |x - y|$. Now imagine the sequence

$$p(0), p(1), \ldots, p(d), p(d+1), \ldots, p\left((d+1)2(2B+1)^{d+1} - 1\right)$$

as being an enumeration of $2(2B+1)^{d+1}$ many $(d+1)$-tuples of numbers in $\{-B, \ldots, B\} \subseteq R$. Each tuple is of the form

$$\left(p\left((d+1)k\right), p\left((d+1)k+1\right), \ldots, p\left((d+1)k+d\right)\right) \in \{-B, \ldots, B\}^{d+1}$$

for $k \in \{0, \ldots, 2(2B+1)^{d+1} - 1\}$. Now, $\{-B, \ldots, B\}^{d+1}$ contains only $(2B+1)^{d+1}$ elements, therefore p must be periodic with its image contained in $p(I)$. Ergo, in $\{0, \ldots, N\}$, it cannot evaluate to a value outside of $\{-B, \ldots, B\}$. □

Proof Lemma 7. Let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be of length s, width B and depth d over R with an offline encryption algorithm $\mathsf{Enc}_{\text{offline}}$. Let $I \subset \mathbb{Z}$ be the interval from lemma 8. We define FE' by setting

$$\begin{aligned}\mathsf{Setup}'(1^\lambda) &:= \mathsf{Setup}(1^\lambda),\\ \mathsf{KeyGen}'(\mathsf{msk}', f) &:= \mathsf{KeyGen}(\mathsf{msk}', f),\\ \mathsf{Dec}'(\mathsf{sk}', \mathsf{ct}') &:= \mathsf{Dec}(\mathsf{sk}, \mathsf{ct})\end{aligned}$$

and by defining Enc' to be of depth d with the following offline algorithm:

$$\begin{aligned}\mathsf{Enc}'_{\text{offline}}(\mathsf{msk}') =\{&\\ &(p_1, \dots, p_s) \leftarrow \mathsf{Enc}_{\text{offline}}(\mathsf{msk}')\\ &\mathsf{if}\ \ \exists x \in I, j \in [s] :\ |p_j(x)|_R > B\\ &\quad \mathsf{return}\ (0, \dots, 0)\\ &\mathsf{else}\\ &\quad \mathsf{return}\ (p_1, \dots, p_s)\\ \}.&\end{aligned}$$

Then, $\mathsf{Enc}'_{\text{offline}}$ is ppt, since I and $[s]$ are of polynomial size. Let $\varepsilon \in \mathsf{negl}(\lambda)$, s.t. we have for all $(x_\lambda)_\lambda$

$$\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)}[\ \exists i \in [s] : |c_i|_R > B \mid c \leftarrow \mathsf{Enc}(\mathsf{msk}, x_\lambda)] \leq \varepsilon(\lambda).$$

We then have

$$\begin{aligned}&\Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)}[\ \exists i \in [s], x \in I : |p_i(x)|_R > B \mid p \leftarrow \mathsf{Enc}_{\text{offline}}(\mathsf{msk})]\\ =&\sum_{x \in I} \Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)}[\ \exists i \in [s] : |p_i(x)|_R > B \mid p \leftarrow \mathsf{Enc}_{\text{offline}}(\mathsf{msk})]\\ =&\sum_{x \in I} \Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)}[\ \exists i \in [s] : |c_i|_R > B \mid c \leftarrow \mathsf{Enc}(\mathsf{msk}, x)] \leq \#I \cdot \varepsilon(\lambda).\end{aligned}$$

Set

$$A := \{p \in \mathsf{supp}(\mathsf{Enc}_{\text{offline}}) \mid \exists i \in [s], x \in I : |p_i(x)|_R > B\}.$$

Then, we have for $\mathsf{msk}, \mathsf{msk}' \leftarrow \mathsf{Setup}(1^\lambda)$

$$\Delta((\mathsf{msk}, \mathsf{Enc}_{\text{offline}}(\mathsf{msk})), (\mathsf{msk}', \mathsf{Enc}'_{\text{offline}}(\mathsf{msk}'))) \leq \Pr_{\mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda)}[\mathsf{Enc}_{\text{offline}}(\mathsf{msk}) \in A] \leq \#I \cdot \varepsilon(\lambda).$$

Since I is of polynomial size, $\#I \cdot \varepsilon(\lambda)$ is negligible. Since we have on the other hand for each $(x_\lambda)_\lambda$

$$\Delta((\mathsf{msk}, \mathsf{Enc}(\mathsf{msk}, x_\lambda)), (\mathsf{msk}', \mathsf{Enc}'(\mathsf{msk}', x_\lambda))) \leq \Delta((\mathsf{msk}, \mathsf{Enc}_{\text{offline}}(\mathsf{msk})), (\mathsf{msk}', \mathsf{Enc}'_{\text{offline}}(\mathsf{msk}'))),$$

it follows that FE is virtually FE'. □

4 Online/Offline Encryption Without Overflows

In this section, we show that private-key encryption schemes of polynomial width that are better than guessing cannot be IND-CPA secure, if their encryption algorithms have a very simple online part in which no arithmetical overflows do occur.

Theorem 6. *Let $d \in \mathbb{N}$ be constant, $N \geq 2d$ and let SKE be a private-key encryption scheme of depth d and width $B \in \mathsf{poly}(\lambda)$ with message space $X_\lambda = \{0, \ldots, N\}$ over $\mathbb{Z}$.*

If SKE is selectively IND-CPA secure, then SKE is useless.

Proof Theorem 6 Part 1. Let SKE be an IND-CPA secure scheme of length s, depth d and width B over $\mathbb{Z}$ for messages $X_\lambda = \{0, \ldots, N\}$. If we define $\mathsf{SKE}' = (\mathsf{Setup}', \mathsf{Enc}', \mathsf{Dec}')$ like in lemma 7, then SKE is virtually SKE'. In particular, SKE' is of the same length and depth and is secure and useless iff SKE is so. Furthermore, SKE' is now strictly of width B, i.e., it never outputs a ciphertext outside of $\{-B, \ldots, B\}^s$. It now suffices to prove that SKE' is useless. ■

To prove theorem 6, we define an adversary which we will show to have a non-negligible advantage against SKE', if SKE' is not useless.

Definition 8. Let $r \in \mathsf{poly}(\lambda)$, $N \geq 2d$ and $s \geq 1$. Set $m = 2r^3$.

We define the following selective adversary $\mathcal{A}$ which plays the IND-CPA security-game in definition 5 with the scheme SKE':

Step 1: The adversary $\mathcal{A}$ draws $y \leftarrow [2d]$ and then, for $b = 0, 1$, submits the following two lists of $3m$ messages each:

$$x_i^b = \begin{cases} 0, & \text{if } i \in \{1, \ldots, m\}, \\ b \cdot y, & \text{if } i \in \{m+1, \ldots, 2m\}, \\ y, & \text{if } i \in \{2m+1, \ldots, 3m\}. \end{cases}$$

He submits two empty lists of possible functions.

Step 2: The adversary $\mathcal{A}$ receives a list of ciphertexts $(\mathsf{ct}'_{x_i^b})_{i=1}^{3m}$. Let $\mathsf{ct}'_{x_i^b, j}$ denote the j-th entry of $\mathsf{ct}'_{x_i^b}$. For $k = 0, 1, 2$ and $j = 1, \ldots, s$ he computes the arithmetical means

$$c_{k,j} := \frac{1}{m} \sum_{i=1+km}^{(k+1)m} (\mathsf{ct}'_{x_i^b, j})^2$$

Step 3: If there is a j s.t. $|c_{2,j} - c_{1,j}| > 2\frac{B}{r}$, the adversary outputs 0. Otherwise, if there is a j s.t. $|c_{0,j} - c_{1,j}| > 2\frac{B}{r}$, he outputs 1. If none of the above requirements should be met, then the adversary outputs a random bit $b' \leftarrow \{0, 1\}$.

The following lemma shows in which cases $\mathcal{A}$ has a non-negligible advantage.

Lemma 9. *Let $r \in \mathsf{poly}(\lambda)$ s.t. $r \geq \lambda$. For a fixed msk', set $\mathsf{CT}'_y = \mathsf{Enc}'(\mathsf{msk}', y)$. The adversary in definition 8 has a non-negligible advantage in the selective IND-CPA game against SKE', if the following probability is non-negligible*

$$\Pr_{\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda)} \left[\exists j \in [s], y^* \in [2d] : \left| \mathbb{E}\left[\left(\mathsf{CT}'_{y^*, j}\right)^2 \right] - \mathbb{E}\left[\left(\mathsf{CT}'_{0, j}\right)^2 \right] \right| > 4\frac{B}{r} \right].$$

Proof. Fix for this proof a master secret key $\mathsf{msk}' \in \mathsf{supp}(\mathsf{Setup}'(1^\lambda))$ and denote by ${\mathsf{CT}'_y}^2$ the distribution of drawing $\mathsf{ct}'_y \leftarrow \mathsf{Enc}'(\mathsf{msk}', y)$ and squaring all its entries. In step 2, $\mathcal{A}$ approximates the means of ${\mathsf{CT}'_0}^2, {\mathsf{CT}'_{b \cdot y}}^2$ and ${\mathsf{CT}'_y}^2$. By $\mathsf{Bounded}$ we denote the event that for each $k = 0, 1, 2$ the distance between c_k and its mean is at most B/r, i.e.

$$\mathsf{Bounded}: \quad \max\left(\left|\left| c_0 - \mathbb{E}\left[{\mathsf{CT}'_0}^2 \right] \right|\right|_\infty, \left|\left| c_1 - \mathbb{E}\left[{\mathsf{CT}'_{b \cdot y}}^2 \right] \right|\right|_\infty, \left|\left| c_2 - \mathbb{E}\left[{\mathsf{CT}'_y}^2 \right] \right|\right|_\infty \right) \leq \frac{B}{r}.$$

Since Enc' always outputs values bounded by B, we have, according to corollary 2, that the probability that event $\mathsf{Bounded}$ will occur is at least $(1-2e^{-r})^{3s} \geq 1-6se^{-r}$. Therefore, for each fixed msk', it follows

$$\Pr\left[\mathcal{A} \text{ fails} \mid b=0\right] \leq \Pr\left[\left|\left|c_0-c_1\right|\right|_\infty > 2\frac{B}{r}\right] + \frac{1}{2} \leq \Pr\left[\neg\mathsf{Bounded}\right] + \frac{1}{2} \leq 6se^{-r} + \frac{1}{2}.$$

Similarly, for each fixed $\mathsf{msk}' \in \mathsf{supp}(\mathsf{Setup}'(1^\lambda))$, we get $\Pr\left[\mathcal{A} \text{ fails} \mid b=1\right] \leq 6se^{-r} + \frac{1}{2}$. Now, assume additionally for msk' that the following event $\mathsf{Seperated}$ does hold

$$\mathsf{Seperated}: \quad \exists y^* \in [2d]: \quad \left|\left|\mathbb{E}\left[{\mathsf{CT}'_0}^2\right] - \mathbb{E}\left[{\mathsf{CT}'_{y^*}}^2\right]\right|\right|_\infty > 4\frac{B}{r}.$$

Let y denote the value drawn by $\mathcal{A}$ in step 1. If $\mathsf{Seperated}$ holds for msk', then

$$\begin{aligned}
&\Pr\left[\mathcal{A} \text{ wins} \mid b=0, y=y^*\right] \\
\geq &\Pr\left[\left|\left|c_2-c_1\right|\right|_\infty > 2\frac{B}{r} \;\middle|\; b=0, y=y^*\right] \\
\geq &\Pr[\mathsf{Bounded}] \cdot \Pr\left[\left|\left|c_2-c_1\right|\right|_\infty > 2\frac{B}{r} \;\middle|\; \mathsf{Bounded}, b=0, y=y^*\right] \\
\geq &(1-6se^{-r}) \cdot 1 = 1-6se^{-r}.
\end{aligned}$$

Similarly, we get $\Pr\left[\mathcal{A} \text{ wins} \mid b=1, y=y^*\right] \geq 1-6se^{-r}$. Therefore, for $\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda)$, we get now

$$\begin{aligned}
&\Pr\left[\mathcal{A} \text{ wins} \mid \mathsf{Seperated}\right] \\
=&\frac{1}{2d}(\Pr\left[\mathcal{A} \text{ wins} \mid \mathsf{Seperated}, y=y^*\right] + \frac{2d-1}{2d}\Pr\left[\mathcal{A} \text{ wins} \mid \mathsf{Seperated}, y\neq y^*\right]) \\
\geq &\frac{1}{2d}(1-6se^{-r}) + \frac{2d-1}{2d}\left(\frac{1}{2}-6se^{-r}\right) \geq \frac{1}{4d}+\frac{1}{2}-6se^{-r}.
\end{aligned}$$

Now, if we set $\varepsilon := \Pr\left[\mathsf{Seperated}\right]$, we have

$$\begin{aligned}
\Pr\left[\mathcal{A} \text{ wins}\right] &= \varepsilon \cdot \Pr\left[\mathcal{A} \text{ wins} \mid \mathsf{Seperated}\right] + (1-\varepsilon) \cdot \Pr\left[\mathcal{A} \text{ wins} \mid \neg\mathsf{Seperated}\right] \\
&\geq \varepsilon\left(\frac{1}{4d}+\frac{1}{2}-6se^{-r}\right) + (1-\varepsilon)\left(\frac{1}{2}-6se^{-r}\right) \\
&= \varepsilon\frac{1}{4d}+\frac{1}{2}+6se^{-r}.
\end{aligned}$$

Since our lemma requires ε to be non-negligible and $r \geq \lambda$, it follows that $\mathcal{A}$ has a non-negligible advantage. □

To conclude the proof of theorem 6, we need to show that the prerequisites of lemma 9 do occur, if SKE' is not useless. In fact, we show a purely mathematical statement in the following which implies the uselessness of SKE', if the prerequisites of lemma 9 are not met. Our statement says that for a distribution of polynomials the means of the squared outputs of the polynomials for $x = 0, \ldots, 2d$ need to be widespread, because, otherwise, it is very unlikely for the sampled polynomials to be non-constant. If the polynomials sampled by $\mathsf{Enc}'_{\text{offline}}(\mathsf{msk}')$ are with overwhelming probability constant, then, of course, the sampled ciphertexts do not carry any information about the encrypted input x.

Lemma 10. *Let $\mathcal{D}$ be a distribution over integer polynomials of degree $d > 0$. If there is a function $\varepsilon = \varepsilon(\lambda)$ s.t. for all $x \in \{1, \ldots, 2d\}$ we have*

$$\left|\mathop{\mathbb{E}}_{p\leftarrow\mathcal{D}}\left[p(x)^2 - p(0)^2\right]\right| \leq \varepsilon,$$

then it follows

$$\Pr_{p\leftarrow\mathcal{D}}\left[\deg p \leq d-1\right] \geq 1-2\varepsilon.$$

Proof. For $p \leftarrow \mathcal{D}$, we set $f(X) := p(X)^2 - p(0)^2$. Then, f is a random integer polynomial of degree $2d$. If we have $p(X) = \sum_{i=0}^{d} a_i X^i$, then the leading coefficient of f is a_d^2. Now, by theorem 3, it follows

$$(2d)! \cdot a_d^2 = \sum_{i=0}^{2d} (-1)^{2d-i} \binom{2d}{i} f(i).$$

Hence

$$\begin{aligned}
\mathop{\mathbb{E}}_{p \leftarrow \mathcal{D}} \left[a_d^2\right] =& \frac{1}{(2d)!} \left| \sum_{i=0}^{2d} (-1)^{2d-i} \binom{2d}{i} \mathop{\mathbb{E}}_{p \leftarrow \mathcal{D}} \left[f(i)\right] \right| \\
\leq& \frac{1}{(2d)!} \sum_{i=0}^{2d} \binom{2d}{i} \left| \mathop{\mathbb{E}}_{p \leftarrow \mathcal{D}} \left[f(i)\right] \right| \\
\leq& \frac{1}{(2d)!} \sum_{i=0}^{2d} \binom{2d}{i} \cdot \varepsilon = \frac{2^{2d}}{(2d)!} \varepsilon \leq 2\varepsilon.
\end{aligned}$$

If we draw $p(X) = \sum_{i=0}^{d} a_i X^i \leftarrow \mathcal{D}$, it follows

$$\Pr\left[\deg p = d\right] = \sum_{i \in \mathbb{Z} \setminus \{0\}} \Pr\left[a_d = i\right] \leq \sum_{i \in \mathbb{Z} \setminus \{0\}} i^2 \cdot \Pr\left[a_d = i\right] = \mathop{\mathbb{E}}_{p \leftarrow \mathcal{D}} \left[a_d^2\right] \leq 2\varepsilon. \qquad \square$$

Lemma 10 already implies that the offline algorithm of an IND-CPA secure encryption scheme of depth d and polynomial width will – with overwhelming probability – sample polynomials of degree $d-1$. In the following theorem, we generalize this observation for arbitrary degrees $d-k$.

Theorem 7. *Let $\mathcal{D}$ be a distribution over integer polynomials of degree d. If there are functions $\varepsilon = \varepsilon(\lambda)$ and $B = B(\lambda)$ s.t. for all $x \in \{1, \ldots, 2d\}$ and $p \in \mathsf{supp}(\mathcal{D})$ we have*

$$\left|p(x)^2 - p(0)^2\right| \leq B^2$$

and

$$\left| \mathop{\mathbb{E}}_{p \leftarrow \mathcal{D}} \left[p(x)^2 - p(0)^2\right] \right| \leq \frac{1}{2}\varepsilon,$$

then it follows for all $k = 0, \ldots, d$

$$\Pr_{p \leftarrow \mathcal{D}} \left[\deg p \leq d - k\right] \geq 1 - (2 + 2B^2)^k \varepsilon.$$

Proof. We show Theorem 7 by using induction over $k = 0, \ldots, d$. While the base case $k = 0$ is trivially true, the case $k = 1$ follows immediately by lemma 10.

For the induction step, let $k \geq 1$ be arbitrary. We will show

$$\Pr[\deg p \leq d - (k+1)] \geq (2B^2 + 1) \Pr[\deg p \leq d - k] - \varepsilon - 2B^2.$$

To this aim, write $p(X) = \sum_{i=0}^{d} a_i X^i$ for $p \leftarrow \mathcal{D}$ and let $\mathcal{D}'$ be $\mathcal{D}$ conditioned on $\deg p \leq d-k$, i.e.

$$\Pr[p \leftarrow \mathcal{D}'] := \Pr[p \leftarrow \mathcal{D} \mid \deg p \leq d - k].$$

Then, we have

$$\begin{aligned}
\Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d - (k+1)] &= \Pr_{p \leftarrow \mathcal{D}}[a_{d-k} = 0 \mid \deg p \leq d - k] \cdot \Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d - k] \\
&= \Pr_{p \leftarrow \mathcal{D}'}[a_{d-k} = 0] \cdot \Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d - k]. \qquad (6)
\end{aligned}$$

We want to apply lemma 10 to show

$$\Pr_{p \leftarrow \mathcal{D}'}[a_{d-k} = 0] \geq 1 - 2 \cdot \frac{\frac{1}{2}\varepsilon + (1 - \Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d - k])B^2}{\Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d - k]}. \qquad (7)$$

For this, we have to show for all $x = 0, \ldots, 2(d-k)$

$$\left| \mathop{\mathbb{E}}_{p \leftarrow \mathcal{D}'} \left[p(x)^2 - p(0)^2 \right] \right| \leq \frac{\frac{1}{2}\varepsilon + (1 - \Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d-k])B^2}{\Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d-k]}. \tag{8}$$

If we set $A := \mathsf{supp}(\mathcal{D}) \setminus \mathsf{supp}(\mathcal{D}')$, then we have for $x \in \{0, \ldots, 2d\}$

$$\begin{aligned}
&\Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d-k] \cdot \left| \mathop{\mathbb{E}}_{p \leftarrow \mathcal{D}'} \left[p(x)^2 - p(0)^2 \right] \right| \\
=&\left| \sum_{p \in \mathsf{supp}(\mathcal{D}')} \Pr\left[p \leftarrow \mathcal{D}\right] \cdot (p(x)^2 - p(0)^2) \right| \\
=&\left| \mathop{\mathbb{E}}_{p \leftarrow \mathcal{D}}[p(x)^2 - p(0)^2] - \sum_{p \in A} \Pr\left[p \leftarrow \mathcal{D}\right] \cdot (p(x)^2 - p(0)^2) \right| \\
\leq&\left| \mathop{\mathbb{E}}_{p \leftarrow \mathcal{D}}[p(x)^2 - p(0)^2] \right| + \left| \sum_{p \in A} \Pr\left[p \leftarrow \mathcal{D}\right] \cdot (p(x)^2 - p(0)^2) \right| \\
\leq&\frac{1}{2}\varepsilon + \Pr_{p \leftarrow \mathcal{D}}[p \in A] \cdot B^2 = \frac{1}{2}\varepsilon + \left(1 - \Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d-k]\right) \cdot B^2.
\end{aligned}$$

By reordering, we get inequality (8). Hence, lemma 10 yields inequality (7). Inserting inequality (7) in equation (6) gives

$$\begin{aligned}
&\Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d - (k+1)] \\
=&\Pr_{p \leftarrow \mathcal{D}'}[a_{d-k} = 0] \cdot \Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d-k] \\
\geq&\left(1 - 2\frac{\left(\frac{1}{2}\varepsilon + (1 - \Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d-k])B^2\right)}{\Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d-k]}\right) \Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d-k] \\
=&\Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d-k] - \varepsilon - 2\left(1 - \Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d-k]\right) B^2 \\
=&(2B^2+1) \Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d-k] - \varepsilon - 2B^2
\end{aligned}$$

The induction hypothesis states

$$\Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d-k] \geq 1 - (2B^2+2)^k \varepsilon.$$

Since $\left((2B^2+1)(2B^2+2)^k + 1\right) \leq (2B^2+2)^{k+1}$, we obtain the claimed inequality:

$$\begin{aligned}
Pr_{p \leftarrow \mathcal{D}}[\deg p \leq d-k] &\geq (2B^2+1)(1 - (2B^2+2)^k\varepsilon) - \varepsilon - 2B^2 \\
&\geq 1 - \left((2B^2+1)(2B^2+2)^k + 1\right)\varepsilon \\
&\geq 1 - (2B^2+2)^{k+1}\varepsilon.
\end{aligned}$$

□

We can now finish the proof of theorem 6.

Proof Theorem 6 Part 2. Let $\mathcal{A}$ be the adversary in definition 8. For $\mathcal{A}$ to have negligible advantage against SKE', according to lemma 9, it is necessary to have for all $r = 4r'B \in \mathsf{poly}(\lambda)$

$$\Pr\left[\forall j \in [s], y \in [2d] : \left| \mathbb{E}\left[\left(\mathsf{CT}'_{y,j}\right)^2\right] - \mathbb{E}\left[\left(\mathsf{CT}'_{0,j}\right)^2\right] \right| \leq \frac{1}{r'}\right] \geq 1 - \mathsf{negl}(\lambda)$$

where we take the probability over $\mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda)$. But now, by theorem 7, we have for each $r \in (2 + 2B^2)^d \cdot \mathsf{poly}(\lambda)$

$$\Pr\left[\forall j \in [s] : \Pr_{(p_1, \ldots, p_s) \leftarrow \mathsf{Enc}'_{\mathrm{offline}}}[\deg p_j = 0] \geq 1 - (2+2B^2)^d \frac{1}{r}\right] \geq 1 - \mathsf{negl}(\lambda).$$

Therefore, the uselessness of SKE' and, in particular, the uselessness of SKE follow. □

5 Online/Offline Encryption With Short Ciphertexts

In section 4, we showed that encryption schemes of constant depth and polynomial width without arithmetic overflows cannot be secure. In this section, we show the same result for encryption schemes of constant depth and polynomial width which may make use of arithmetic overflows but have short ciphertexts. We do so by transforming such schemes to encryption schemes without arithmetic overflows. I.e., if the ciphertexts are of short width, we can transform their encryption algorithm to one of constant depth over $\mathbb{Z}$ by using a simple multiplication trick. As before, throughout this section, let λ denote the security parameter and let $B = B(\lambda), d = d(\lambda)$ and $N = N(\lambda)$ be arbitrary variables depending on λ. Let $s \in \mathsf{poly}(\lambda)$. Additionally, introduce a modulus variable $q = q(\lambda)$. We prove in this section the following theorem:

Theorem 8. *Let q be a prime, $N \geq d+1$ and let SKE_q be a private-key encryption scheme of depth d and width B over $\mathbb{Z}_q$ for messages $X_\lambda = \{0, \ldots, N\}$ s.t.*

$$2(d+1)^2 \cdot (d!)^3 \cdot d^d \cdot N^d \cdot B \leq q - 1.$$

If SKE_q is selectively IND-CPA secure, then there exists a virtual adversarial transformation to an encryption scheme SKE of depth d and width $(d!)^2 B$ over $\mathbb{Z}$ for messages $X_\lambda = \{0, \ldots, N\}$ which preserves selective IND-CPA security and – in both directions – correctness, being better than guessing and uselessness.

Theorem 8 and theorem 6 imply together the following impossibility result:

Corollary 3. *Let q be a prime and let SKE_q be a private-key encryption scheme of depth d and width B for messages $x = 0, \ldots, N$ over $\mathbb{Z}_q$ s.t. $N \geq 2d$ and*

$$2(d+1)^2 \cdot (d!)^3 \cdot d^d \cdot N^d \cdot B \leq q - 1.$$

If SKE_q is selectively IND-CPA secure, $B \in \mathsf{poly}(\lambda)$ and $d \in \mathbb{N}$ constant, then SKE_q is useless.

Proof. Because of theorem 8, there is an IND-CPA secure private-key encryption scheme SKE over $\mathbb{Z}$ of polynomial width $(d!)^2 B$ and constant depth $d \in \mathbb{N}$ for messages $X_\lambda = \{0, \ldots, N\}$ which is useless iff SKE_q is useless. Since $N \geq 2d$, SKE is useless according to theorem 6. □

To prove theorem 8, let $q > 2$ be a prime and define a map

$$\iota : \mathbb{Z}_q \to \{-\frac{q-1}{2}, \ldots, 0, \ldots, \frac{q-1}{2}\} \subset \mathbb{Z}$$

by setting

$$\iota(a \bmod q) := a + zq \quad \text{where } z \in \mathbb{Z} \text{ s.t. } |a + zq| = |a \bmod q|\,.$$

Then, ι preserves absolute values and we have

$$\iota(a \bmod q) \bmod q = a \bmod q.$$

One first idea for proving theorem 8 could be to just apply ι component-wise to each ciphertext, i.e. treat each ciphertext modulo q as it would be an integer vector. Technically, we would replace Enc by $\iota \circ \mathsf{Enc}$. While $\iota \circ \mathsf{Enc}$ would be indeed of length s and width B over $\mathbb{Z}$, it is not clear, if it would be of depth d over $\mathbb{Z}$. To make this precise, for $p \in \mathbb{Z}_q[X]$, we denote by $I(p \bmod q)$ the coefficient-wise application of ι, i.e.

$$I\left(\sum_{i=0}^{d} a_i X^i \bmod q\right) := \sum_{i=0}^{d} \iota(a_i \bmod q) X^i.$$

Then, we have the equation $I(p \bmod q) \bmod q = p \bmod q$ again. Now, for $\iota \circ \mathsf{Enc}$ to be of depth d over $\mathbb{Z}$, we would need a suitable offline algorithm. We could, for example, take $I \circ \mathsf{Enc}_{\text{offline}}$ as candidate. If p is a polynomial over $\mathbb{Z}_q$ sampled by $\mathsf{Enc}_{\text{offline}}$, we would then need the following kind of equality for all $x \in X_\lambda$

$$\iota(p(x) \bmod q) = I(p \bmod q)(x). \tag{9}$$

While equation (9) holds for polynomials p with small coefficients, it does not hold in general. Therefore, we need to apply minor changes to the polynomials sampled by $\mathsf{Enc}_{\text{offline}}$ as we will see later. To this end, consider the Vandermonde matrix for the tuple $(0, 1, \ldots, d)$

$$V := ((i-1)^{j-1})_{i,j=1,\ldots,d+1} = \begin{pmatrix} 1 & 0 & 0 & \ldots & 0 \\ 1 & 1 & 1 & \ldots & 1 \\ 1 & 2 & 4 & \ldots & 2^d \\ \vdots & & & & \vdots \\ 1 & d & d^2 & \ldots & d^d \end{pmatrix} \in \mathbb{Z}^{(d+1)\times(d+1)}.$$

We can deduce the coefficients of a polynomial by applying V^{-1} to its output values. However, V^{-1} has very large entries modulo q, therefore we use the following integer *quasi-inverse* W with bounded entries.

Lemma 11. *There exists an integer matrix $W \in \mathbb{Z}^{(d+1)\times(d+1)}$ whose entries are bounded by $(d!)^3 d^d$, s.t. $V \cdot W = W \cdot V = (d!)^2 \cdot \mathrm{Id}_{(d+1)\times(d+1)}$.*

Proof. Let $w_0, \ldots, w_d$ be integer polynomials of degree d. We denote their coefficients by $w_{i,j}$, i.e.

$$w_i(X) = \sum_{i=0}^{d} w_{i,j} X^j.$$

If we set

$$W := \begin{pmatrix} w_{0,0} & \ldots & w_{d,0} \\ \vdots & \ddots & \vdots \\ w_{0,d} & \ldots & w_{d,d} \end{pmatrix},$$

we have

$$VW = \begin{pmatrix} w_0(0) & \ldots & w_d(0) \\ \vdots & \ddots & \vdots \\ w_0(d) & \ldots & w_d(d) \end{pmatrix}.$$

If we set

$$\widehat{w}_j(X) := \left(\prod_{i=0, i\neq j}^{d} (X - i) \right)$$

and

$$w_j(X) := (d!)^2 \frac{1}{\widehat{w}_j(j)} \cdot \widehat{w}_j(X),$$

then each w_j is an integer polynomial with

$$w_j(i) = \begin{cases} (d!)^2, & \text{if } j = i, \\ 0, & \text{if } j \neq i. \end{cases}$$

Hence, $VW = (d!)^2 \cdot \mathrm{Id}_{(d+1)\times(d+1)}$. To bound the entries of W, we have to bound the coefficients of $\sum_{i=0}^{d} \widehat{w}_{i,j} X^j := \widehat{w}_i(X)$. We can compute $\widehat{w}_{i,d-j}$ by

$$\widehat{w}_{i,d-j} = \sum_{\substack{\{k_1,\ldots,k_j\} \subseteq \{0,\ldots,d\}\setminus\{i\} \\ k_1 < \ldots < k_j}} (-1)^j \cdot k_1 \cdots k_j.$$

Therefore

$$|\widehat{w}_{i,d-j}| = \sum_{\substack{\{k_1,\ldots,k_j\} \subseteq \{0,\ldots,d\}\setminus\{i\} \\ k_1 < \ldots < k_j}} k_1 \cdots k_j \leq \sum_{\substack{\{k_1,\ldots,k_j\} \subseteq \{0,\ldots,d\}\setminus\{i\} \\ k_1 < \ldots < k_j}} d^j \leq \binom{d}{j} d^j.$$

Since

$$w_j(X) = \frac{(d!)^2}{\widehat{w}_j(j)} \cdot \widehat{w}_j(X) \leq (d!)^2 \widehat{w}_j(X),$$

we get

$$|w_{i,j}| \leq \binom{d}{j} d^j (d!)^2 \leq (d!)^3 d^d. \qquad \square$$

Lemma 12. *Let $q > 2$ be a prime, set $c = (d!)^2$ and let $p \in \mathbb{Z}_q[X]$ be a polynomial of degree d. Furthermore, let $N \geq d+1$. If we have for all $x = 0, \ldots, d$*

$$|p(x) \bmod q| \leq \frac{q-1}{2(d+1)^2 \cdot (d!)^3 \cdot d^d \cdot N^d},$$

then we have for all $x = 0, \ldots, N$

$$I(c \cdot p \bmod q)(x) = \iota(c \cdot p(x) \bmod q).$$

Proof. It is clear that we have for any integer polynomial p and any $x \in \mathbb{Z}$

$$I(c \cdot p \bmod q)(x) \bmod q = c \cdot p(x) \bmod q = \iota(c \cdot p(x) \bmod q) \bmod q.$$

Therefore, in our case, it suffices to show that the absolute value of $I(c \cdot p \bmod q)(x)$ is bounded by $\frac{q-1}{2}$, since $\iota(c \cdot p(x) \bmod q)$ is a value of $\{-\frac{q-1}{2}, \ldots, \frac{q-1}{2}\}$ which differs from $I(c \cdot p \bmod q)(x)$ only by a value in $q\mathbb{Z}$.

Let $p(X) = \sum_{i=0}^{d} a_i X^i \in \mathbb{Z}_q[X]$ and set $a = (a_0, \ldots, a_d) \in \mathbb{Z}_q^{d+1}$ to be the column vector of $p's$ coefficients. Then, we have

$$V \cdot a \bmod q = \begin{pmatrix} 1 & 0 & 0 & \ldots & 0 \\ 1 & 1 & 1 & \ldots & 1 \\ 1 & 2 & 4 & \ldots & 2^d \\ \vdots & & & & \vdots \\ 1 & d & d^2 & \ldots & d^d \end{pmatrix} \cdot \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_d \end{pmatrix} \bmod q = \begin{pmatrix} p(0) \\ p(1) \\ p(2) \\ \vdots \\ p(d) \end{pmatrix} \bmod q.$$

Let $W = (w_{i,j})_{i,j} \in \mathbb{Z}^{(d+1)\times(d+1)}$ be the quasi-inverse of V from lemma 11. Since

$$WVa = ca \mod q,$$

we have for each a_i

$$c \cdot a_i \bmod q = \sum_{i=0}^{d} w_{i,j} p(j) \bmod q.$$

In particular, we have now

$$|c \cdot a_i \bmod q| = \left| \sum_{i=0}^{d} w_{i,j} p(j) \bmod q \right| \leq \sum_{i=0}^{d} |w_{i,j}| \cdot |p(j) \bmod q|.$$

Set

$$B := \max_{x=0,\ldots d} |p(x) \bmod q| \leq \frac{q-1}{2(d+1)^2 \cdot (d!)^3 \cdot d^d \cdot N^d}.$$

Since each $|w_{i,j}|$ is bounded by $(d!)^3 d^d$ and each $|p(j) \bmod q|$ is bounded by B, we get

$$|c \cdot a_i \bmod q| \leq \sum_{i=0}^{d} |w_{i,j}| \cdot |p(j) \bmod q| \leq \sum_{i=0}^{d} (d!)^3 d^d B = (d+1)(d!)^3 d^d B.$$

Therefore, we have for all $x = 0, \ldots, N$

$$\begin{aligned}
|I(c \cdot p \bmod q)(x)| &= \left| \sum_{i=0}^{d} \iota\left(c \cdot a_i \bmod q\right) x^i \right| \\
&\leq \sum_{i=0}^{d} \left| \iota\left(c \cdot a_i \bmod q\right) x^i \right| \\
&\leq \sum_{i=0}^{d} \left| \iota\left(c \cdot a_i \bmod q\right) \right| \cdot \left| x^i \right| \\
&\leq \sum_{i=0}^{d} (d+1)(d!)^3 d^d B \cdot |x|^i \\
&\leq (d+1)(d!)^3 d^d B \cdot \left(\sum_{i=0}^{d} N^i \right) \\
&\leq (d+1)(d!)^3 d^d B \cdot (d+1) N^d \leq \frac{q-1}{2}.
\end{aligned}$$

Ergo, the claim follows. □

Proof Theorem 8. Because of lemma 7, we can – by using the same argument we used in the first part of the proof of theorem 6 – w.l.o.g. assume that the encryption algorithm of $\mathsf{SKE}_q = (\mathsf{Setup}_q, \mathsf{Enc}_q, \mathsf{Dec}_q)$ never outputs a ciphertext whose entries modulo q are not bounded by B. Set

$$c := (d!)^2 \in \mathbb{Z}, \quad h := c^{-1} \bmod q \in \mathbb{Z}_q$$

and define a scheme $\mathsf{SKE} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{Dec})$ over $\mathbb{Z}$ by applying the following adversarial transformation to SKE_q:

$$\begin{aligned}
\mathsf{Setup}(1^\lambda) &:= \mathsf{Setup}_q(1^\lambda), \\
\mathsf{Enc}(\mathsf{msk}, x) &:= \iota(c \cdot \mathsf{Enc}_q(\mathsf{msk}, x) \bmod q), \\
\mathsf{Dec}(\mathsf{msk}, \mathsf{ct}) &:= \mathsf{Dec}_q(\mathsf{msk}, (h \cdot \mathsf{ct} \bmod q)).
\end{aligned}$$

It is clear that SKE_q is correct, better than guessing (resp. useless) iff SKE is correct, better than guessing (resp. useless), since we have

$$(h \cdot (\iota(c \cdot \mathsf{ct} \bmod q)) \bmod q) = (h \cdot (c \cdot \mathsf{ct}) \bmod q) = \mathsf{ct} \bmod q.$$

Since SKE_q is IND-CPA secure and the above transformations are adversarial, SKE is IND-CPA secure.

It remains to show that Enc is an encryption algorithm of depth d and width cB over $\mathbb{Z}$. Now, for each $(\mathsf{ct}_1, \ldots, \mathsf{ct}_s) \leftarrow \mathsf{Enc}_q(\mathsf{msk}, x)$, we have

$$|\iota(c \cdot \mathsf{ct}_j \bmod q)| = |c \cdot \mathsf{ct}_j \bmod q| \leq c \cdot |\mathsf{ct}_j \bmod q| \leq cB,$$

therefore Enc is of width cB over $\mathbb{Z}$. To show that Enc is of depth d we have to give a feasible offline algorithm $\mathsf{Enc}_{\text{offline}}$ for $\mathsf{Enc} = \iota(c \cdot \mathsf{Enc}_q)$. This is done by setting

$$\mathsf{Enc}_{\text{offline}}(\mathsf{msk}) := I(c \cdot \mathsf{Enc}_{\text{offline},q}(\mathsf{msk}) \bmod q).$$

Let $x \in \{0, \ldots, N\}$. If we fix the randomness r of $\mathsf{Enc}(\mathsf{msk}, x, r)$ and set

$$\begin{aligned}
(p_1, \ldots, p_s) &:= \mathsf{Enc}_{\text{offline},q}(\mathsf{msk}, r), \\
(p'_1, \ldots, p'_s) &:= \mathsf{Enc}_{\text{offline}}(\mathsf{msk}, r),
\end{aligned}$$

then

$$\begin{aligned}\mathsf{Enc}(\mathsf{msk}, x, r) =& \iota(c \cdot \mathsf{Enc}_q(\mathsf{msk}, x, r) \bmod q) \\ =& (\iota(c \cdot p_1(x) \bmod q), \ldots, \iota(c \cdot p_s(x) \bmod q)) \\ \overset{(*)}{=}& (I(c \cdot p_1 \bmod q)(x), \ldots, I(c \cdot p_s \bmod q)(x)) \\ =& (p'_1(x), \ldots, p'_s(x)),\end{aligned}$$

where equation $(*)$ follows from lemma 12. Therefore, $\mathsf{Enc}(\mathsf{msk}, x)$ is of depth d. □

6 Lattice-Based Function-Hiding Functional Encryption

In this section, let $n(\lambda) \geq 1$ be a polynomial in λ and let

$$q(\lambda) > p(\lambda) \geq N(\lambda) \geq 1$$

for all λ. Further, let $X_\lambda = \{0, \ldots, p\}^n$, $Y_\lambda = \{0, \ldots, p\}$ and let $(F_\lambda)_\lambda$ be a function family which contains (besides other functions) the zero-function $0 \in F_\lambda$ – which maps each $x \in X_\lambda$ to zero – and the projection $\pi_1 \in F_\lambda$ – which maps each $x \in X_\lambda$ to its first coordinate.

Let $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be a functional encryption scheme for $(F_\lambda)_\lambda$ of depth d_1 and length s over $\mathbb{Z}_q$ and let $d_2 \in \mathbb{N}$ be a constant s.t. each secret key $\mathsf{sk} \in \mathsf{supp}(\mathsf{KeyGen})$ is a polynomial in $\mathbb{Z}_q[X_1, \ldots, X_s]$ of total degree $\leq d_2$ with

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = \lceil \mathsf{sk}(\mathsf{ct}) / \lfloor q/p \rfloor \rfloor \,.$$

Finally, set $m = \binom{s+d_2}{d_2}$. We prove in this section the following theorem:

Theorem 9. *If q is a prime and FE is selectively $(m+1)$-bounded function-hiding IND-CPA secure and correct, then there exists an adversarial transformation from FE to a private-key encryption scheme of depth $d := d_1 \cdot d_2$, width $\lfloor q/p \rfloor$ and length m over $\mathbb{Z}_q$ for messages $x = 0, \ldots, N$ which is selectively IND-CPA secure and better than guessing.*

Corollary 4 (Impossibility Result)**.** *Assume that q is a prime, d_1 is constant and $\frac{q}{p}$ is bounded by a polynomial in λ and that for almost all $\lambda \in \mathbb{N}$ we have*

$$p(\lambda) \geq (d+1)^2 \cdot 2^{d+1} \cdot (d!)^3 \cdot d^{2d}.$$

Then, FE cannot be both selectively $(m+1)$-bounded function-hiding IND-CPA secure and correct.

Proof. Assume that FE is both and set $N = 2d$. Because of theorem 9, we can transform FE to a private-key encryption scheme over $\mathbb{Z}_q$ with depth d and width $B := \lfloor q/p \rfloor$ for messages $X'_\lambda = \{0, \ldots, 2d\}$ which is IND-CPA secure and better than guessing. Then, we have

$$B = \left\lfloor \frac{q}{p} \right\rfloor \leq \frac{q-1}{p} \leq \frac{q-1}{2(d+1)^2 \cdot (d!)^3 \cdot d^d \cdot (2d)^d}.$$

Now, according to corollary 3, this encryption scheme must be useless and therefore cannot be better than guessing. In particular, FE cannot be correct. □

We prove theorem 9 by applying adversially three transformations to FE. First, we relinearize the ciphertexts and secret keys s.t. decryption becomes evaluating a scalar product, dividing by $\lfloor q/p \rfloor$ and rounding down. Second, we draw m secret keys $v_1, \ldots, v_m \leftarrow \mathsf{KeyGen}'(\mathsf{msk}, 0)$ for the zero-function and replace a ciphertext ct' with a vector of decryption noises $\langle \mathsf{ct}' \mid v_i \rangle$. Because of decryption correctness, each noise value must be small; therefore, we get a new ciphertext of small width. By using sufficiently many secret keys, we can ensure that the new ciphertext contains enough information s.t. the probability of a correct decryption becomes high enough. We will not always be able to decrypt correctly, but we show that we are still better than guessing by $\frac{1}{m}$. In fact, this is implied by lemma 13 which states that a secret key of a non-zero function must sufficiently resemble a secret key of the zero-function. As a last step, we convert the current FE scheme into a private-key encryption scheme for messages $x \in \{0, \ldots, N\}$ which is better than guessing and of small width over $\mathbb{Z}_q$. Since all transformations can be applied by an adversary, the scheme stays IND-CPA secure (however, we lose some security in the second transformation step, since we have to ask for m secret keys). If we started with a FE scheme of constant depth, then the final scheme will also be of constant depth.

Proof Theorem 9 Step 1. As a first step, we relinearize the ciphertexts and secret keys of FE. Note that each polynomial $\mathsf{sk} \in \mathbb{Z}_q[X_1, \ldots, X_s]$ of total degree $\leq d_2$ can be written as a vector of its coefficients. This yields a linear transformation

$$\Phi : \{\mathsf{sk} \in \mathbb{Z}_q[X_1, \ldots, X_s] \mid \deg \mathsf{sk} \leq d_2\} \longrightarrow \mathbb{Z}_q^{\binom{s+d_2}{d_2}}.$$

On the other hand, there is a polynomial map $\Phi^+ : \mathbb{Z}_q^s \longrightarrow \mathbb{Z}_q^m$ of degree d_2 which maps each vector to a vector of different products of its entries s.t. we have for all $\mathsf{sk} \in \mathbb{Z}_q[X_1, \ldots, X_s]$ of total degree $\leq d_2$ and all $\mathsf{ct} \in \mathbb{Z}_q^s$

$$\mathsf{sk}(\mathsf{ct}) = \left\langle \Phi\left(\mathsf{sk}\right) \mid \Phi^+\left(\mathsf{ct}\right) \right\rangle. \tag{10}$$

Now, we define a new scheme $\mathsf{FE}' = (\mathsf{Setup}', \mathsf{KeyGen}', \mathsf{Enc}', \mathsf{Dec}')$ by setting

$$\begin{aligned} \mathsf{Setup}'(1^\lambda) &:= \mathsf{Setup}(1^\lambda), & \mathsf{KeyGen}'(\mathsf{msk}', f) &:= \Phi\left(\mathsf{KeyGen}(\mathsf{msk}', f)\right), \\ \mathsf{Enc}'(\mathsf{msk}', x) &:= \Phi^+\left(\mathsf{Enc}(\mathsf{msk}', x)\right), & \mathsf{Dec}'(\mathsf{sk}', \mathsf{ct}') &:= \left\lceil \left\langle \mathsf{sk}' \mid \mathsf{ct}' \right\rangle / \lfloor q/p \rfloor \right\rfloor. \end{aligned}$$

Applying Φ and Φ^+ together forms an adversarial transformation, therefore FE' is $(m+1)$-fh-IND-CPA secure. Because of equation (10), FE' is correct. Further, Enc' is of depth $d := d_1 \cdot d_2$ and its outputs are vectors of length $m = \binom{s+d_2}{d_2}$. ■

Lemma 13. *For each sampleable $(f_\lambda)_\lambda \in (F_\lambda)_\lambda$ there is an $\varepsilon \in \mathsf{negl}(\lambda)$ s.t.*

$$\Pr\left[\mathsf{sk}'_f \in \mathsf{span}_{\mathbb{Z}_q}\{v_1, \ldots, v_m\} \;\middle|\; \begin{array}{c} \mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda) \\ v_1, \ldots, v_m \leftarrow \mathsf{KeyGen}'(\mathsf{msk}, 0) \\ \mathsf{sk}'_f \leftarrow \mathsf{KeyGen}'(\mathsf{msk}', f_\lambda) \end{array} \right] \geq \frac{1}{m+1} - \varepsilon(\lambda).$$

Proof. Lemma 4 states

$$P_1 := \Pr\left[\mathsf{sk}'_0 \in \mathsf{span}_{\mathbb{Z}_q}\{v_1, \ldots, v_m\} \;\middle|\; \begin{array}{c} \mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda) \\ v_1, \ldots, v_m \leftarrow \mathsf{KeyGen}'(\mathsf{msk}', 0) \\ \mathsf{sk}'_0 \leftarrow \mathsf{KeyGen}'(\mathsf{msk}', 0) \end{array} \right] \geq 1 - \frac{1}{m+1}.$$

Consider an adversary $\mathcal{A}$ who plays the IND-CPA game from definition 5 against FE' and works as follows:

Step 1: For $b = 0, 1$ and $i = 1, \ldots, m+1$, the adversary sets

$$g_i^b := \begin{cases} 0, & \text{if } i \leq m \text{ or } b = 0, \\ f_\lambda, & \text{if } i = m+1 \text{ and } b = 1. \end{cases}$$

and submits two empty lists of possible inputs and two lists of possible functions $(g_i^0)_{i=1}^{m+1}, (g_i^1)_{i=1}^{m+1}$.

Step 2: After receiving $(\mathsf{sk}'_{g_i^b})_{i=1}^{m+1}$, $\mathcal{A}$ computes $V := \mathsf{span}_{\mathbb{Z}_q}\left\{\mathsf{sk}'_{g_1^b}, \ldots, \mathsf{sk}'_{g_m^b}\right\}$.

Step 3: The adversary outputs 0, if $\mathsf{sk}'_{g_{m+1}^b} \in V$, and 1 otherwise.

If we set

$$P_2 := \Pr\left[\mathsf{sk}'_f \in \mathsf{span}_{\mathbb{Z}_q}\{v_1, \ldots, v_m\} \;\middle|\; \begin{array}{c} \mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda), \\ v_1, \ldots, v_{m-1} \leftarrow \mathsf{KeyGen}'(\mathsf{msk}', 0), \\ \mathsf{sk}'_f \leftarrow \mathsf{KeyGen}'(\mathsf{msk}', f_\lambda) \end{array} \right],$$

then we can compute the advantage of $\mathcal{A}$ by

$$\varepsilon := \Pr[\mathcal{A} \text{ wins} \mid b = 0] + \Pr[\mathcal{A} \text{ wins} \mid b = 1] - 1 = P_1 + (1 - P_2) - 1 = P_1 - P_2.$$

ε is negligible, since FE' is $(m+1)$-fh-IND-CPA secure. Therefore

$$P_2 = P_1 - \varepsilon(\lambda) \geq \frac{1}{m+1} - \varepsilon(\lambda).$$ □

Proof Theorem 9 Step 2. Let $\mathsf{FE}' = (\mathsf{Setup}', \mathsf{KeyGen}', \mathsf{Enc}', \mathsf{Dec}')$ be a correct and $(m+1)$-fh-IND-CPA secure functional encryption scheme where Enc' is of depth d and length m over $\mathbb{Z}_q$. Let furthermore Dec' be computed by

$$\mathsf{Dec}'(\mathsf{sk}', \mathsf{ct}') = \left\lceil \left\langle \mathsf{sk}' \mid \mathsf{ct}' \right\rangle / \lfloor q/p \rfloor \right\rfloor.$$

We now adversarially transform FE' to a functional encryption scheme FE'' for the same functionality which is 1-fh-IND-CPA secure, better than guessing and whose encryption algorithm has depth d, width $\lfloor q/p \rfloor$ and length m over $\mathbb{Z}_q$.

In the IND-CPA game against FE', our adversary first queries m secret keys

$$v_1, \ldots, v_m \leftarrow \mathsf{KeyGen}'(\mathsf{msk}', 0)$$

for the zero function and then makes use of the algorithm $\mathcal{B}$ described in theorem 5 to compute $V, A, A^+ \leftarrow \mathcal{B}(v_1, \ldots, v_m)$ s.t.

$$V = \mathsf{span}_{\mathbb{Z}_q}\{v_1, \ldots, v_m\}$$

and $A \in \mathbb{Z}_q^{m \times k}, A^+ \in \mathbb{Z}_q^{k \times m}$ are matrices with

$$V = A \cdot \mathbb{Z}_q^k \quad \text{and} \quad A \cdot A^+ v = v \text{ for all } v \in V.$$

After our adversary queried m secret keys, FE' remains 1-fh-IND-CPA secure. However, by doing so, the adversary gained the additional data V, A, A^+ with which he can transform FE' to $\mathsf{FE}'' = (\mathsf{Setup}'', \mathsf{KeyGen}'', \mathsf{Enc}'', \mathsf{Dec}'')$ by setting:

$\mathsf{Setup}''(1^\lambda) := \mathsf{Setup}'(1^\lambda)$

$\mathsf{Enc}''(\mathsf{msk}'', x) := A^T \cdot \mathsf{Enc}'(\mathsf{msk}'', x)$

$\mathsf{KeyGen}''(\mathsf{msk}'', f)$:
$\mathsf{sk}'_f \leftarrow \mathsf{KeyGen}'(\mathsf{msk}', f)$
if $\mathsf{sk}'_f \in V$
  $\mathsf{sk}''_f := A^+ \cdot \mathsf{sk}'_f$
else
  $\mathsf{sk}''_f := \bot$
return sk''_f

$\mathsf{Dec}''(\mathsf{sk}'', \mathsf{ct}'')$:
if $\mathsf{sk}'' = \bot$
  $y \leftarrow \{0, \ldots, p\}$
else
  $y \leftarrow \mathsf{Dec}'(\mathsf{sk}'', \mathsf{ct}'')$
return y

FE'' has the following properties:

- **Security:** The above changes can be applied by an adversary while he plays the IND-CPA game from definition 5. Therefore, FE'' is 1-fh-IND-CPA secure, since our adversary has to query m secret keys for the zero function which does not leak any information about encrypted messages.

- **Depth and Length:** Since the transformation of the encryption algorithm is done by multiplication with the matrix $A^T \in \mathbb{Z}_q^{k \times m}$, the depth of the encryption algorithm does not change. Furthermore, Enc'' is of length[4] $k \leq m$ over $\mathbb{Z}_q$.

- **Width:** We have to show that Enc'' is of width $\lfloor q/p \rfloor$. To this end, let $(x_\lambda)_\lambda \in (X_\lambda)_\lambda$, draw $\mathsf{msk}'' \leftarrow \mathsf{Setup}''(1^\lambda)$, $\mathsf{ct}'' \leftarrow \mathsf{Enc}''(\mathsf{msk}'', x_\lambda)$ and fix a component ct''_i of $\mathsf{ct}'' = (\mathsf{ct}''_1, \ldots, \mathsf{ct}''_k) \in \mathbb{Z}_q^k$. Note that the columns of the matrix $A = (v_{j_1} | \ldots | v_{j_k})$ are some of the vectors $v_1, \ldots, v_m \leftarrow \mathsf{KeyGen}'(\mathsf{msk}', 0)$ according to theorem 5. Since $\mathsf{ct}'' = A^T \mathsf{ct}'$ for some $\mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x_\lambda)$, there is, because of the correctness of FE', an $\varepsilon_0 \in \mathsf{negl}(\lambda)$ s.t. for all $(x_\lambda)_\lambda \in (X_\lambda)_\lambda$

$$\begin{aligned}\Pr\left[|\mathsf{ct}''_i| \leq \left\lfloor \frac{q}{p} \right\rfloor\right] &= \Pr\left[|v_{j_i}^T \cdot \mathsf{ct}'| \leq \left\lfloor \frac{q}{p} \right\rfloor\right] \geq \Pr\left[\left\lfloor \frac{v_{j_i}^T \cdot \mathsf{ct}'}{\lfloor q/p \rfloor} \right\rfloor = 0\right] \\ &= \Pr\left[\mathsf{Dec}'(v_{j_i}, \mathsf{ct}') = 0 \;\middle|\; \begin{array}{l} \mathsf{msk}' \leftarrow \mathsf{Setup}'(1^\lambda) \\ v_{j_i} \leftarrow \mathsf{KeyGen}'(\mathsf{msk}', 0), \\ \mathsf{ct}' \leftarrow \mathsf{Enc}'(\mathsf{msk}', x) \end{array}\right] \geq 1 - \varepsilon(\lambda)\end{aligned}$$

[4]Note that k is not fixed but rather a random variable. However, this is not a problem, since we can always pad the output of Enc'' to be of length m over $\mathbb{Z}_q$.

where in the first three terms we take the randomness over the computation of msk'' and ct''. Therefore, Enc'' is of width $\lfloor q/p \rfloor$.

– **Better than Guessing:** It remains to show that FE'' is better than guessing.

Fix $(x_\lambda)_\lambda \in (X_\lambda)_\lambda$ and a samplable $(f_\lambda)_\lambda \in (F_\lambda)_\lambda$ and draw

$$\begin{aligned}\mathsf{msk}'' &\leftarrow \mathsf{Setup}''(1^\lambda),\\ \mathsf{sk}''_f &\leftarrow \mathsf{KeyGen}''(\mathsf{msk}'', f_\lambda),\\ \mathsf{ct}''_x &\leftarrow \mathsf{Enc}''(\mathsf{msk}'', x_\lambda).\end{aligned}$$

Then, we have

$$\begin{aligned}&\Pr\left[\mathsf{Dec}''(\mathsf{sk}''_f, \mathsf{ct}''_x) = f(x)\right]\\ =&\Pr\left[\mathsf{Dec}''(\mathsf{sk}''_f, \mathsf{ct}''_x) = f(x) \mid \mathsf{sk}''_f = \bot\right] \cdot \Pr\left[\mathsf{sk}''_f = \bot\right]\\ +&\Pr\left[\mathsf{Dec}''(\mathsf{sk}''_f, \mathsf{ct}''_x) = f(x) \mid \mathsf{sk}''_f \neq \bot\right] \cdot \Pr\left[\mathsf{sk}''_f \neq \bot\right]\\ =&\frac{1}{p+1} \cdot \Pr\left[\mathsf{sk}''_f = \bot\right] + \Pr\left[\mathsf{Dec}''(\mathsf{sk}''_f, \mathsf{ct}''_x) = f(x) \mid \mathsf{sk}''_f \neq \bot\right] \cdot \Pr\left[\mathsf{sk}''_f \neq \bot\right].\end{aligned}$$

Now, we have

$$\mathsf{sk}''_f \neq \bot \iff \mathsf{sk}'_f \in V.$$

Because of lemma 13, the probability for this is at least $\frac{1}{m+1} - \varepsilon_1$ for some $\varepsilon_1 \in \mathsf{negl}(\lambda)$. If $\mathsf{sk}'_f \in V$, we have

$$\begin{aligned}\mathsf{Dec}''(\mathsf{sk}''_f, \mathsf{ct}''_x) = \mathsf{Dec}'(\mathsf{sk}''_f, \mathsf{ct}''_x) &= \left\lceil \frac{\langle \mathsf{sk}''_f \mid \mathsf{ct}''_x \rangle}{\lfloor q/p \rfloor} \right\rfloor = \left\lceil \frac{\langle A^+ \mathsf{sk}'_f \mid A^T \mathsf{ct}'_x \rangle}{\lfloor q/p \rfloor} \right\rfloor\\ &= \left\lceil \frac{\langle AA^+ \mathsf{sk}'_f \mid \mathsf{ct}'_x \rangle}{\lfloor q/p \rfloor} \right\rfloor = \mathsf{Dec}'(\mathsf{sk}'_f, \mathsf{ct}'_x).\end{aligned}$$

The last term equals $f_\lambda(x_\lambda)$ with probability at least $1 - \varepsilon_2$ for some $\varepsilon_2 \in \mathsf{negl}(\lambda)$. Now, let λ be big enough s.t. $1 - \varepsilon_2(\lambda) \geq \frac{1}{p(\lambda)+1}$, then

$$\begin{aligned}&\Pr\left[\mathsf{Dec}''(\mathsf{sk}''_f, \mathsf{ct}''_x) = f(x)\right] \qquad (11)\\ =&\frac{1}{p+1} \cdot \Pr\left[\mathsf{sk}''_f = \bot\right] + \Pr\left[\mathsf{Dec}''(\mathsf{sk}''_f, \mathsf{ct}''_x) = f(x) \mid \mathsf{sk}''_f \neq \bot\right] \cdot \Pr\left[\mathsf{sk}''_f \neq \bot\right]\\ \geq&\frac{1}{p+1} \cdot (1 - \Pr\left[\mathsf{sk}''_f \neq \bot\right]) + (1 - \varepsilon_2) \cdot \Pr\left[\mathsf{sk}''_f \neq \bot\right]\\ =&\frac{1}{p+1} + \Pr\left[\mathsf{sk}''_f \neq \bot\right] \left(1 - \varepsilon_2 - \frac{1}{p+1}\right)\\ \geq&\frac{1}{p+1} + \left(\frac{1}{m+1} - \varepsilon_1\right) \left(1 - \varepsilon_2 - \frac{1}{p+1}\right)\\ \geq&\frac{1}{p+1} + \frac{p}{(m+1)(p+1)} - \mathsf{negl}(\lambda).\end{aligned}$$

Therefore, FE'' is better than guessing by $\frac{p}{(m+1)(p+1)}$. ■

Since $(F_\lambda)_\lambda$ contains the projection onto the first coordinate, there is a straightforward way to adversially transform FE'' to a private encryption scheme over $\mathbb{Z}_q$ with width $\lfloor q/p \rfloor$ and depth d which is better than guessing and selectively IND-CPA secure. For this purpose set $\widetilde{X}_\lambda = \{0, \ldots, N(\lambda)\}$.

Proof Theorem 9 Step 3. Let $\mathsf{FE}'' = (\mathsf{Setup}'', \mathsf{KeyGen}'', \mathsf{Enc}'', \mathsf{Dec}'')$ be the functional encryption scheme of the preceding step. Then, FE'' is 1-fh-IND-CPA secure, better than guessing and of

depth d and width $B := \lfloor q/p \rfloor$ over $\mathbb{Z}_q$. Additionally, FE'' has the special property that for all samplable $(f_\lambda)_\lambda$ there is an $\varepsilon \in \mathsf{negl}(\lambda)$, s.t. we have for all $(x_\lambda)_\lambda$

$$\Pr_{\mathsf{msk}'' \leftarrow \mathsf{Setup}''(1^\lambda)} \left[\mathsf{Dec}''(\mathsf{sk}''_f, \mathsf{ct}''_x) = f_\lambda(x_\lambda) \;\middle|\; \begin{array}{l} \mathsf{sk}''_f \leftarrow \mathsf{KeyGen}''(\mathsf{msk}'', f_\lambda) \\ \mathsf{ct}''_x \leftarrow \mathsf{Enc}''(\mathsf{msk}'', x_\lambda) \\ \mathsf{sk}''_f \neq \bot \end{array} \right] \geq 1 - \varepsilon(\lambda).$$

We adversarially transform FE'' to a private-key encryption scheme $\mathsf{SKE}''' = (\mathsf{Setup}''', \mathsf{Enc}''', \mathsf{KeyGen}''', \mathsf{Dec}''')$ of depth d and width B over $\mathbb{Z}_q$ for the message space $\widetilde{X}_\lambda$ which is IND-CPA secure and better than guessing. For this end set:

$$\begin{aligned} \mathsf{Setup}'''(1^\lambda) &:= \mathsf{Setup}''(1^\lambda) \\ \mathsf{Enc}'''(\mathsf{msk}''', x) &:= \mathsf{Enc}''(\mathsf{msk}''', (x, 0 \ldots, 0)) \\ \mathsf{KeyGen}'''(\mathsf{msk}''', \mathrm{Id}_{\widetilde{X}_\lambda}) &:= \mathsf{KeyGen}''(\mathsf{msk}''', \pi_1) \end{aligned}$$

and

$\mathsf{Dec}'''(\mathsf{sk}''', \mathsf{ct}''')$:
if $\mathsf{sk}''' = \bot$
  $y \leftarrow \{0, \ldots, N\}$
else
  $y \leftarrow \mathsf{Dec}''(\mathsf{sk}''', \mathsf{ct}''')$
return y

Note that this adversarial transformation is the only one in this work, where we have two functional encryption schemes for different functionalities. Now, SKE''' is IND-CPA secure, because FE'' is 1-fh-IND-CPA secure (in fact, FE'' being 0-fh-IND-CPA secure would already suffice). Enc''' is of depth d and width B over $\mathbb{Z}_q$, since Enc'' is so. The computations marked by the number (11) in the preceding transformation step show – mutatis mutandis – that SKE''' is better than guessing by $\frac{N}{(m+1)\cdot(N+1)}$. □

References

[ABB10] Shweta Agrawal, Dan Boneh, and Xavier Boyen, *Efficient lattice (H)IBE in the standard model*, EUROCRYPT 2010 (Henri Gilbert, ed.), LNCS, vol. 6110, Springer, Heidelberg, May / June 2010, pp. 553–572.

[ABDP15] Michel Abdalla, Florian Bourse, Angelo De Caro, and David Pointcheval, *Simple functional encryption schemes for inner products*, PKC 2015 (Jonathan Katz, ed.), LNCS, vol. 9020, Springer, Heidelberg, March / April 2015, pp. 733–751.

[ABKW19] Michel Abdalla, Fabrice Benhamouda, Markulf Kohlweiss, and Hendrik Waldner, *Decentralizing inner-product functional encryption*, PKC 2019, Part II (Dongdai Lin and Kazue Sako, eds.), LNCS, vol. 11443, Springer, Heidelberg, April 2019, pp. 128–157.

[ACF+18] 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*, CRYPTO 2018, Part I (Hovav Shacham and Alexandra Boldyreva, eds.), LNCS, vol. 10991, Springer, Heidelberg, August 2018, pp. 597–627.

[ACF+19] Shweta Agrawal, Michael Clear, Ophir Frieder, Sanjam Garg, Adam O'Neill, and Justin Thaler, *Ad hoc multi-input functional encryption*, Cryptology ePrint Archive, Report 2019/356, 2019, `https://eprint.iacr.org/2019/356`.

[AFV11] Shweta Agrawal, David Mandell Freeman, and Vinod Vaikuntanathan, *Functional encryption for inner product predicates from learning with errors*, ASIACRYPT 2011 (Dong Hoon Lee and Xiaoyun Wang, eds.), LNCS, vol. 7073, Springer, Heidelberg, December 2011, pp. 21–40.

[AJ15] Prabhanjan Ananth and Abhishek Jain, *Indistinguishability obfuscation from compact functional encryption*, CRYPTO 2015, Part I (Rosario Gennaro and Matthew J. B. Robshaw, eds.), LNCS, vol. 9215, Springer, Heidelberg, August 2015, pp. 308–326.

[AJS18] Prabhanjan Ananth, Aayush Jain, and Amit Sahai, *Indistinguishability obfuscation without multilinear maps: io from lwe, bilinear maps, and weak pseudorandomness*, Cryptology ePrint Archive, Report 2018/615, 2018, `https://eprint.iacr.org/2018/615`.

[ALS16] Shweta Agrawal, Benoît Libert, and Damien Stehlé, *Fully secure functional encryption for inner products, from standard assumptions*, CRYPTO 2016, Part III (Matthew Robshaw and Jonathan Katz, eds.), LNCS, vol. 9816, Springer, Heidelberg, August 2016, pp. 333–362.

[AR17] Shweta Agrawal and Alon Rosen, *Functional encryption for bounded collusions, revisited*, TCC 2017, Part I (Yael Kalai and Leonid Reyzin, eds.), LNCS, vol. 10677, Springer, Heidelberg, November 2017, pp. 173–205.

[AV19] Prabhanjan Ananth and Vinod Vaikuntanathan, *Optimal bounded-collusion secure functional encryption*, Cryptology ePrint Archive, Report 2019/314, 2019, `https://eprint.iacr.org/2019/314`.

[BCFG17] Carmen Elisabetta Zaira Baltico, Dario Catalano, Dario Fiore, and Romain Gay, *Practical functional encryption for quadratic functions with applications to predicate encryption*, CRYPTO 2017, Part I (Jonathan Katz and Hovav Shacham, eds.), LNCS, vol. 10401, Springer, Heidelberg, August 2017, pp. 67–98.

[BGG+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*, EUROCRYPT 2014 (Phong Q. Nguyen and Elisabeth Oswald, eds.), LNCS, vol. 8441, Springer, Heidelberg, May 2014, pp. 533–556.

[BGV12] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan, *(Leveled) fully homomorphic encryption without bootstrapping*, ITCS 2012 (Shafi Goldwasser, ed.), ACM, January 2012, pp. 309–325.

[BJK15] Allison Bishop, Abhishek Jain, and Lucas Kowalczyk, *Function-hiding inner product encryption*, ASIACRYPT 2015, Part I (Tetsu Iwata and Jung Hee Cheon, eds.), LNCS, vol. 9452, Springer, Heidelberg, November / December 2015, pp. 470–491.

[BRS13] Dan Boneh, Ananth Raghunathan, and Gil Segev, *Function-private identity-based encryption: Hiding the function in functional encryption*, CRYPTO 2013, Part II (Ran Canetti and Juan A. Garay, eds.), LNCS, vol. 8043, Springer, Heidelberg, August 2013, pp. 461–478.

[BV11] Zvika Brakerski and Vinod Vaikuntanathan, *Efficient fully homomorphic encryption from (standard) LWE*, 52nd FOCS (Rafail Ostrovsky, ed.), IEEE Computer Society Press, October 2011, pp. 97–106.

[BV15] Nir Bitansky and Vinod Vaikuntanathan, *Indistinguishability obfuscation from functional encryption*, 56th FOCS (Venkatesan Guruswami, ed.), IEEE Computer Society Press, October 2015, pp. 171–190.

[CDG+18] Jérémy Chotard, Edouard Dufour Sans, Romain Gay, Duong Hieu Phan, and David Pointcheval, *Decentralized multi-client functional encryption for inner product*, ASIACRYPT 2018, Part II (Thomas Peyrin and Steven Galbraith, eds.), LNCS, vol. 11273, Springer, Heidelberg, December 2018, pp. 703–732.

[CHKP10] David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert, *Bonsai trees, or how to delegate a lattice basis*, EUROCRYPT 2010 (Henri Gilbert, ed.), LNCS, vol. 6110, Springer, Heidelberg, May / June 2010, pp. 523–552.

[DDM16] Pratish Datta, Ratna Dutta, and Sourav Mukhopadhyay, *Functional encryption for inner product with full function privacy*, PKC 2016, Part I (Chen-Mou Cheng, Kai-Min Chung, Giuseppe Persiano, and Bo-Yin Yang, eds.), LNCS, vol. 9614, Springer, Heidelberg, March 2016, pp. 164–195.

[ESLL19] Muhammed F. Esgin, Ron Steinfeld, Joseph K. Liu, and Dongxi Liu, *Lattice-based zero-knowledge proofs: New techniques for shorter and faster constructions and applications*, CRYPTO 2019, Part I (Alexandra Boldyreva and Daniele Micciancio, eds.), LNCS, vol. 11692, Springer, Heidelberg, August 2019, pp. 115–146.

[Gay20] Romain Gay, *A new paradigm for public-key functional encryption for degree-2 polynomials*, Cryptology ePrint Archive, Report 2020/093, 2020, `https://eprint.iacr.org/2020/093`.

[GKP94] Ronald L. Graham, Donald E. Knuth, and Oren Patashnik, *Concrete mathematics: A foundation for computer science*, 2nd ed., Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1994.

[GKW17] Rishab Goyal, Venkata Koppula, and Brent Waters, *Lockable obfuscation*, 58th FOCS (Chris Umans, ed.), IEEE Computer Society Press, October 2017, pp. 612–621.

[GVW13] Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee, *Attribute-based encryption for circuits*, 45th ACM STOC (Dan Boneh, Tim Roughgarden, and Joan Feigenbaum, eds.), ACM Press, June 2013, pp. 545–554.

[Hoe63] Wassily Hoeffding, *Probability inequalities for sums of bounded random variables*, Journal of the American Statistical Association **58** (1963), no. 301, 13–30.

[HW14] Susan Hohenberger and Brent Waters, *Online/offline attribute-based encryption*, PKC 2014 (Hugo Krawczyk, ed.), LNCS, vol. 8383, Springer, Heidelberg, March 2014, pp. 293–310.

[JLMS19] Aayush Jain, Huijia Lin, Christian Matt, and Amit Sahai, *How to leverage hardness of constant-degree expanding polynomials overa $\mathbb{R}$ to build $i\mathcal{O}$*, EUROCRYPT 2019, Part I (Yuval Ishai and Vincent Rijmen, eds.), LNCS, vol. 11476, Springer, Heidelberg, May 2019, pp. 251–281.

[Lin17] Huijia Lin, *Indistinguishability obfuscation from SXDH on 5-linear maps and locality-5 PRGs*, CRYPTO 2017, Part I (Jonathan Katz and Hovav Shacham, eds.), LNCS, vol. 10401, Springer, Heidelberg, August 2017, pp. 599–629.

[LPR10] Vadim Lyubashevsky, Chris Peikert, and Oded Regev, *On ideal lattices and learning with errors over rings*, EUROCRYPT 2010 (Henri Gilbert, ed.), LNCS, vol. 6110, Springer, Heidelberg, May / June 2010, pp. 1–23.

[LT17] Huijia Lin and Stefano Tessaro, *Indistinguishability obfuscation from trilinear maps and block-wise local PRGs*, CRYPTO 2017, Part I (Jonathan Katz and Hovav Shacham, eds.), LNCS, vol. 10401, Springer, Heidelberg, August 2017, pp. 630–660.

[Reg05] Oded Regev, *On lattices, learning with errors, random linear codes, and cryptography*, 37th ACM STOC (Harold N. Gabow and Ronald Fagin, eds.), ACM Press, May 2005, pp. 84–93.

[WZ17] Daniel Wichs and Giorgos Zirdelis, *Obfuscating compute-and-compare programs under LWE*, 58th FOCS (Chris Umans, ed.), IEEE Computer Society Press, October 2017, pp. 600–611.