On the Complexity of Homomorphic Encryption

LEE, Chin Ho

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of
Master of Philosophy
in
Computer Science and Engineering

The Chinese University of Hong Kong
September 2013

Thesis/Assessment Committee

Professor Pak Ching Lee (Chair)
Professor Andrej Bogdanov (Thesis Supervisor)
Professor Lap Chi Lau (Committee Member)
Professor Alon Rosen (External Examiner)

Abstract

On the complexity of homomorphic encryption

by

LEE, Chin Ho

Master of Philosophy

Department of Computer Science and Engineering

The Chinese University of Hong Kong

Homomorphic encryption was proposed by Rivest, Adleman, and Dertouzos over three decades ago as a mechanism for secure delegation of computation to an honest but curious server. This thesis examines the complexity-theoretic aspects of homomorphic encryption. In particular we are interested in the complexity and provable security of homomorphic encryption. Our results include the following:

- We propose a new homomorphic encryption scheme based on the hardness of decoding under independent random noise from certain affine families of codes. This candidate is later shown to be insecure.
- We give evidence that encryption schemes that support homomorphic evaluation are inherently more complex than ordinary ones. We show that secure homomorphic evaluation of any non-trivial functionality of sufficiently many inputs with respect to any CPA secure encryption scheme cannot be implemented by constant depth, polynomial size circuits, i.e. in the class AC^0. We view this as evidence that encryption schemes that support homomorphic evaluation are inherently more complex than ordinary ones.
- We show that public-key bit encryption schemes which support weak (i.e., compact) homomorphic evaluation of any sufficiently "sensitive" collection of functions cannot be proved message indistinguishable beyond AM$\cap$coAM via general (adaptive) reductions, and beyond statistical zero-knowledge via reductions of constant query complexity.

摘要

同態加密的複雜度

李展浩

香港中文大學

計算機科學與工程學系

哲學碩士

同態加密是由 Rivest、Adleman 和 Dertouzos 於三十年前提出，作爲安全地委任一個誠實遵守協議但同時好奇解密的服務器作運算的一個機制。本論文研究同態加密計算在複雜性理論上的問題，我們特別關注同態加密的複雜度及它的可證明安全性，我們的研究結果包括：

- 我們提出一個新的同態加密方案，這個方案是基於對帶有獨立隨機雜訊的編碼仿射族解碼的困難，它後來被證明是不安全的。
- 我們提供證據去證明支持同態運算的加密方案本質上較普通加密方案複雜，我們證明任何足夠多輸入的非平凡函數的同態運算並不能被一個常量深度和多項式大小的電路（即 AC^0 類）去實現，這可以看作同態加密較普通加密複雜的證據。
- 我們證明如果一個公開密鑰加密方案支持足夠「靈敏」的函數集合的弱（即緊）同態運算，它的信息不可分辨性不能透過一般（自適應）歸約法去證明在 $AM \cap coAM$ 以外，及不能透過常量查詢複雜度的歸約法去證明在統計零知識（SZK）以外。

Acknowledgements

Foremost, I would like to thank Andrej, Lap Chi, Patrick and Alon for serving on my thesis committee.

Above all, this thesis would have been impossible without the incredible support and guidance of Andrej, who has made the last two years of my life more wonderful than I could have imagined. I am truly grateful for his suggestion on choosing this intriguing research topic; for his intuitive explanation of many concepts; for his tremendous contribution to every single work appeared in this thesis, and for his endless supply of coffee, which I failed to turn into theorems. His modesty, way of thinking and eloquent writing have become examples for me to pursue in the future.

My exposure in other areas of theoretical computer science would not have been possible without the regular theory seminars and theory lunches. Thanks to Andrej, Lap Chi and Shengyu for organizing these events, and Chandra, ChiuChiu, Chiwang, Hollis, Joseph, Siyao and many others in the theory group for their talks and enlightening discussions. Another prop goes to those who are also my officemates for keeping the windowless office energetic and entertaining.

Thanks to Benny Applebaum, Cheuk Ting Li, Chris Peikert and Alon Rosen for many useful comments on the work appeared in this thesis.

During my undergraduate studies, I was very fortunate to be a member of the CUHK ACM programming team. From there I acquired skills that are applicable in my graduate studies. I would like to thank Bill, ChiuChiu, Ding Qian, GagGuy, Gary, Hackson, Joe, KN, Lin Jian, Leo, Wai Hon, Peng Hao, Peter, Zhang Qi and many others, with whom I spent hours honing my problem solving (and AOC) skills every week. I would like to extend my gratitude to Lap Chi for giving me the opportunity to be part of the team.

My thanks also go to my students, especially Annie, Chu Mei, Colin, Benny, Daniel, Doris, Jamie, Philip, Roy, Sunny, Wendy and Yau, for all the laughters and fun in the past two years.

Thanks to my grandma for always supporting my decision, and my best friend, Cindy, for giving me countless encouragement and advice that have changed many aspects of my life.

Finally, I thank my beloved Sabrina for her endless patience and endurance.

Contents

Chapter 1

Introduction

This thesis studies the complexity-theoretic aspects of homomorphic encryption, with the focus on its complexity and provable security. In particular, it attempts to answer the following questions:

- How complex is homomorphic encryption? Can it be efficient? Does there exist an homomorphic encryption that admits a parallel implementation?

- How secure is homomorphic encryption? Is it as secure as an ordinary encryption scheme? Can its security be based on NP-complete problems?

Before delving into these problems, let us give an overview of homomorphic encryption and the related problems.

1.1 Overview for homomorphic encryption

Homomorphic encryption was proposed in the seminal work of Rivest, Adleman, and Dertouzos [RAD78] over three decades ago as a mechanism for secure delegation of computation to an honest but curious server. It allows to take encryptions of some messages and some functionality f, and produces a ciphertext that decrypts to the evaluation of f on the messages using only public information.

In the RSA [RSA78] encryption scheme, a public key is generated as follows: First, choose two random prime numbers p and q and consider their product $N = pq$, then output a number e such that $\gcd(e, \phi(N)) = 1$. To encrypt a message m, the encryption algorithm simply outputs $m^e \bmod N$. It is straightforward to see that the scheme is

multiplicative homomorphic, because

$$\mathbf{Enc}_{PK}(m_1) \cdot \mathbf{Enc}_{PK}(m_2) = m_1^e m_2^e = \mathbf{Enc}_{PK}(m_1 m_2).$$

However, since there is no randomness involved in the encryption algorithm, the RSA scheme is not semanticlly secure. While there exist several variants of RSA that achieve semantic security, it remains a open question that whether one can construct an additive or multiplicative homomorphic encryption scheme based on the RSA assumption.

Some partial progress [GM84, Gam85, Pai99] was made on constructing homomorphic encryption schemes based on other assumptions over time. The security of these candidates relies on the hardness of some computational number theoretic problems such as the quadratic residuosity problem and the decisional Diffie-Hellman problem [Bon98]. However, none of these candidates are *fully homomorphic* as they only support homomorphic evaluation of restricted classes of functions. For example, the Goldwasser-Micali (GM) cryptosystem [GM84] only supports homomorphic evaluation of parity. In the GM cryptosystem, the public key is a pair (N, y), where N is again the product of two random prime integers, y is a random quadratic non-residue modulo N. To encrypt a bit b, the encryption algorithm chooses r from $\mathbb{Z}_N$ uniformly at random and outputs $r^2 y^b \bmod N$. Notice that

$$\mathbf{Enc}_{PK}(b_1) \cdot \mathbf{Enc}_{PK}(b_2) = (r_1^2 y^{b_1}) \cdot (r_2^2 y^{b_2}) = (r_1 r_2)^2 y^{b_1 + b_2} = \mathbf{Enc}_{PK}(b_1 \oplus b_2)$$

and so this scheme is additive homomorphic over $\mathbb{Z}_2$.

Other homomorphic encryption schemes include the ElGamal cryptosystem [Gam85] and the Paillier encryption scheme [Pai99], which are only multiplicative homomorphic and additive homomorphic, respectively. The Boneh-Goh-Nissim cryptosystem [BGN05] supports homomomorphic additions and one level of multiplication. Only a few years ago, the first fully homomorphic encryption (FHE) schemes were proposed, starting with the breakthrough work of Gentry [Gen09a, Gen09b] in 2009. Since then, several such schemes have been proposed [vDGHV10, BV11, GH11, BGV12]. However, current implementations of these encryption schemes are not practical.

Most of the existing FHE schemes rely their security on the learning with error (LWE) problem introduced by Regev [Reg05]. In the LWE problem, given a dimension n, a modulus q and an error distribition χ over $\mathbb{Z}_q$, one can ask for arbitrary many samples in the form of $(a, \langle a, s \rangle + e)$, where $a \sim \mathbb{Z}_q^n$ and $e \sim \chi$, and is required to recover s. The LWE assumption says that any probabilistic polynomial time adversary cannot

recover s with non-negligible probability. The decision variant of the LWE assumption (dLWE) assumes the distribution $(a, \langle a, s\rangle + e)$ is pseudorandom, that is, no probabilistic polynomial time adversary can distinguish the distribution from uniformly random with non-negligible probability. It is known that LWE and dLWE are as hard as each other up to certain settings of parameters [MM11, BLP+13]. Currently, the best known algorithm for LWE with any B-bounded distribution χ (i.e. $\Pr_{e\sim\chi}[|e| > B] \leq 1/\mathrm{poly}(n)$) runs in time $2^{\tilde{O}(n/\log(q/B))}$ [LP11].

1.1.1 Applications of homomorphic encryption

One immediate application of homomomorphic encryption is cloud computing. Suppose in a university every student's academic record is encrypted using an ordinary encryption scheme and is stored in a cloud server. If one is interested to find out the average grades of the algorithm course last year, one has to download the entire database of the CS students, decrypt the encrypted data, and carry out the statistical calculation on the data.

In contrast, if the data is encrypted using a homomorphic encryption scheme that supports evaluating some specific class of functions (e.g. statistical average) homomorphically, then given only the public key and the encrypted data, the cloud server can output the encryption of average grades of the course, without performing any decryption during the whole process.

As a result, homomorphic encryption allows the cloud users to, on the one hand, outsource both the storage and computation to the cloud server. This is particularly useful when users do not have the resources to perform the computation themselves. On the other hand, homomorphic encryption also addresses the privacy concerns raised by the users, because the computation does not reveal any information of their data.

Other applications of homomorphic encryption include constructions of several cryptographic protocols for multiparty computation, private information retrieval, eletronic voting and zero-knowledge protocol. While there exist more elegant and efficient constructions of these protocols, the constructions using homomorphic encryption are usually conceptually simpler.

As an example, we now give a construction of a two party computation protocol using homomorphic encryption [IP07].

Two party computation. Alice has an input x and Bob has an input y. Their goal is to compute $f(x, y)$ for some function f without revealing their input to the other one.

Let $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ be a public key homomorphic encryption scheme that supports homomorphic evaluation of f. Consider the following protocol:

1. Alice generates a public/secret key pair (PK, SK) and sends $(PK, C_x = \mathbf{Enc}_{PK}(x))$ to Bob.

2. Bob computes $C_y = \mathbf{Enc}_{PK}(y)$ and evaluates f on x and y homomorphically using PK, C_x, C_y, and the homomorphic evaluator of f. Then he sends the output of the evaluation C to Alice.

3. Alice decrypts the ciphertext C to get $f(x, y)$ and sends it to Bob.

Since everything in step two is encrypted, it is not difficult to see that provided the encryption is CPA secure, and Alice and Bob do not deviate from the protocol, after executing the protocol both of them can get hold of $f(x, y)$, without obtaining any information about the other's input.

1.2 Complexity of cryptographic tasks

A central objective in the theory of cryptography is to classify the relative complexity of various cryptographic tasks. One common way of arguing that task B is of comparable easiness to task A is to give a black-box implementation of B using A as a primitive. Notable examples include the construction of pseudorandom generators from one-way permutations [GL89] and one-way functions [HILL99, HRV10].

But how should we argue that task B is "more complex" than task A? In the generic setting, one looks for the existence of a black-box separation [IR89, RTV04], or a lower bound on the query complexity of a black-box reduction [GT00]. However such black box impossibility results are not always a good indicator of the relative complexity of the two tasks in the real world (under suitable complexity assumptions). For example, although collision-resistant hash functions cannot be constructed from one-way functions in a black-box manner [Sim98], both objects have simple, local (NC^0) implementations under standard assumptions [AIK07].

An alternative way to argue that task B is more complex than task A is to provide a concrete complexity model in which one can implement A (under plausible assumptions), but not B. For example, Applebaum et al. [AIK07] show that under plausible complexity assumptions, nontrivial pseudorandom generators can be implemented in the complexity class NC^0. However, it is not difficult to see that this class does not contain

pseudorandom functions; in fact, Linial, Mansour, and Nisan [LMN93] show that pseudorandom functions cannot be implemented even in AC^0. Taken together, these results may be viewed as concrete evidence that pseudorandom functions are more complex than pseudorandom generators, despite the existence of a black-box reduction [GGM86] and the lack of lower bounds on the complexity of such reductions [MV11].

1.3 Provable security in cryptography

If P equals NP then computationally secure encryption is impossible. Is the converse true?

Despite considerable efforts, there is no candidate encryption scheme whose security can be plausibly reduced to the worst-case hardness of some NP-complete problem. Neither is there conclusive evidence that rules out constructions of secure encryption schemes from NP-complete problems, although several obstacles have been pointed out over the years.

Restricting the encryption Brassard [Bra79] shows that no public-key encryption scheme can be proved secure beyond $\mathrm{NP} \cap \mathrm{coNP}$, but under the implicit assumption that every public key-ciphertext pair (queried by the reduction) can be decrypted uniquely. Goldreich and Goldwasser [GG98] argue that this assumption is unrealistic by giving examples of encryption schemes that do not satisfy it. They show that the conclusion holds under the relaxed assumption that invalid queries to the decryption oracle can be efficiently certified as such. (If the reduction is randomized, the limitation weakens to $\mathrm{AM} \cap \mathrm{coAM}$.)

Goldreich and Goldwasser warn that these assumptions are unrealistic as they do not apply to many known proofs of security. Bogdanov and Trevisan [BT06] point out the following example of Even and Yacobi [EY80]. They construct a public key encryption scheme and show how to solve an NP-hard problem using a distinguishing oracle. Their notion of security is unrealistic, as they require a perfect distinguishing oracle. However, their example illustrates that the restrictions imposed by Brassard and Goldreich and Goldwasser do not capture the difficulty of basing cryptography on NP hardness.

Akavia, Goldreich, Goldwasser, and Moshkovitz [AGGM06] rule out reductions from NP-complete problems to inverting one-way functions (the basis of private-key encryption) assuming that sizes of preimage sets are worst-case certifiable in NP. The same considerations apply to their argument. There are natural examples of conjectured one-way functions (for example, Goldreich's function [Gol00]) not known to satisfy the

aforementioned assumptions.

Restricting the reduction Another line of works makes restrictive assumptions about the type of reduction used to prove NP-hardness. Feigenbaum and Fortnow [FF93] show that a decision problem cannot be proven NP-hard on average (unless the polynomial hierarchy collapses) by a reduction that is non-adaptive and each of its queries is uniformly distributed. Bogdanov and Trevisan [BT06] obtain the same conclusion without restricting the distribution of queries, but still under non-adaptive reductions. More precisely, they show that if there is a non-adaptive reduction from a decision problem L to a problem in distributional NP, then L must be in AM/poly $\cap$ coAM/poly. In particular their result applies to the problem of inverting a one-way function. For this important case, Akavia et al. improve the limitation to AM $\cap$ coAM, also assuming the reduction is non-adaptive.

Haitner, Mahmoody, and Xiao [HMX10] show that collision resistant hash functions and statistically hiding commitments cannot be proved secure beyond AM $\cap$ coAM via reductions that make a constant number of rounds of calls to the adversary.

Lattice-based cryptography provides examples of encryption schemes whose insecurity would imply worst-case solutions to conjectured hard problems, like finding short vectors in lattices [Ajt96]. The reduction of Regev [Reg09], which gives the most efficient cryptosystems of this kind with a proof of security (against quantum algorithms), is adaptive. For certain settings of parameters, these cryptosystems support homomorphic evaluation of a bounded class of functionalities (and general functionalities under additional security assumptions) [Gen09b, vDGHV10, BV11].

1.4 Results and organization

We begin this thesis by giving a definition of homomorphic encryption in the next chapter. In Chapter 3 and 4, we examine the complexity of homomorphic encryption. Our focus is on the possibility of efficient implementation of homomorphic encryption. In Chapter 5, we study the provably security for homomorphic encryption. In Chapter 6, we discuss some open problems for further research.

Chapter 3 – Homomorphic encryption from codes. We attempt to equip known cryptosystems that admit efficient implementation with homomorphism. Owing to the simplicity of encryption, the scheme of Applebaum, Barak and Wigderson [ABW10] is a natural starting point for this study. To this end, we propose a new homomorphic

encryption scheme based on the hardness of decoding under independent random noise from certain affine families of codes. However, this candidate is shown to be insecure by the independent work of Brakerski [Bra13] and Gauthier et al. [GOT12]. We discuss their attacks in detail at the end of the chapter.

Chapter 4 – On the depth complexity of homomorphic encryption schemes. We show that secure homomorphic evaluation of any non-trivial functionality of sufficiently many inputs with respect to any CPA secure encryption scheme cannot be implemented by constant depth, polynomial size circuits, i.e. in the class AC^0. In contrast, we observe that certain previously studied encryption schemes (with quasipolynomial security) can be implemented in AC^0. We view this as evidence that encryption schemes that support homomorphic evaluation are inherently more complex than ordinary ones.

Chapter 5 – Limits of provable security for homomorphic encryption. We show that public-key bit encryption schemes which support homomorphic evaluation of any sufficiently "sensitive" collection of functions cannot be proved message indistinguishable beyond $\mathrm{AM} \cap \mathrm{coAM}$ via general (adaptive) reductions, and beyond statistical zero-knowledge via reductions of constant query complexity. Examples of sensitive collections include parities, majorities, and the class consisting of all AND and OR functions.

Our techniques also give a method for converting a strong homomorphic evaluator for essentially any boolean function (except the trivial ones, the NOT function, and the AND and OR functions) into a rerandomization algorithm: This is a procedure that converts a ciphertext into another ciphertext which is statistically close to being independent and identically distributed with the original one. Our transformation preserves negligible statistical error.

Most of the material in this thesis also appears in the work [BL11, BL12, BL13].

Chapter 2

Definition of homomorphic evaluation

In this chapter we give a definition of homomorphic evaluation which will be used throughout this thesis. There are several variants of the definition of what it means for an algorithm H to homomorphically evaluate a given functionality f. A fairly weak requirement is that a homomorphic evaluator for $f(m_1, \ldots, m_k)$ should take as inputs encryptions of $m_1, \ldots, m_k$ and output a ciphertext that decrypts to $f(m_1, \ldots, m_k)$.

To account for the possibility that the encryption scheme itself may produce incorrect encryptions with some probability, we will allow for the evaluation algorithm to err on some fraction of the encryptions.

Definition 2.1. Let $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ be a private-key encryption scheme over message set Σ and ciphertext set Ξ. We say H is a *homomorphic evaluator* of $f : \Sigma^* \to \Sigma$ with error δ if (1) the output length of H is bounded by a function that depends only on the security parameter and (2) for all n and $m \in \Sigma^n$ in the domain of f,

$$\Pr[\mathbf{Dec}_{SK}(H(\mathbf{Enc}_{SK}(m_1, R_1), \ldots, \mathbf{Enc}_{SK}(m_n, R_n))) = f(m)] \geq 1 - \delta,$$

where $SK \sim \mathbf{Gen}$ is a uniformly chosen secret key and $R_1, \ldots, R_n$ are independent random seeds.

In the public-key setting, we are given an encryption scheme $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ and require that

$$\Pr[\mathbf{Dec}_{SK}(H_{PK}(\mathbf{Enc}_{PK}(m_1, R_1), \ldots, \mathbf{Enc}_{PK}(m_n, R_n))) = f(m)] \geq 1 - \delta,$$

where $(PK, SK) \sim \mathbf{Gen}$ is a random key pair.

We point out condition (1) in this definition is necessary in the context of ruling out the existence of trivial homomorphic evaluators. When k is much smaller than n, condition (2) alone allows for plausible encryption schemes that admit trivial homomorphic evaluators, by "outsourcing" the homomorphic evaluation to the decryption algorithm. For example suppose that the meaningful portion of an encryption is only captured in the first n/k bits of the ciphertext. Then the homomorphic evaluator can simply copy the meaningful portion of its k encryptions in non-overlapping parts of the output. Upon seeing a ciphertext of this form, the decryption algorithm can easily compute the value $f(m_1, \ldots, m_k)$ by first decrypting the ciphertext corresponding to each of the k encryptions and then evaluating f.

A stronger notion of homomorphic evaluation will be defined in Chapter 5.

Chapter 3

Homomorphic encryption from codes

3.1 Introduction

In this chapter we propose a new way to achieve homomorphic encryption based on codes rather than lattices. In both code and lattice based cryptosystems, encryptions are obtained by applying a transformation to an input and adding some noise. The two differ in the noise model. In lattice-based encryption schemes, the noise is required to be of bounded magnitude. In code-based schemes, the noise vector must have sufficiently small hamming weight, but is otherwise unrestricted.

Our main result is a construction of a homomorphic public-key encryption scheme from a code-based public-key encryption scheme with some special properties. The code-based scheme which is the base of our construction is new. We arrived at it by combining the structure of encryptions of the local cryptosystem of Applebaum, Barak, and Wigderson [ABW10] with a "key scrambling" idea of the McEliece cryptosystem [McE78]. We also provide a definitional framework for homomorphic encryption that may be useful elsewhere.

The security assumption of the scheme is shown to be false by the independent works of Brakerski [Bra13], and Gauthier, Otmani and Tillich [GOT12]. We present their attacks in Section 3.6.

We begin by discussing the proposed scheme and give evidence in favor of its security. The design is motivated by certain algebraic requirements that enable the implementation of homomorphic operations. We defer the discussion of these special properties to Section 3.2.

3.1.1 The base cryptosystem K

The ciphertexts in our cryptosystem are n-bit vectors over $\mathbb{F}_q$, where q is a power of a prime. Three additional parameters that enter the description of the cryptosystem are the amount of randomness r used in the encryption, the size s of the secret key, and the noise distribution $\tilde{\eta}$ over $\mathbb{F}_q$. We will discuss the relationships between these parameters shortly. Conjecture 3.1 at the end of this section summarizes the conclusion of this discussion. The message set of our encryption scheme is the set $\mathbb{F}_q$.

Public-key encryption scheme K

Key generation: Choose a uniformly random subset $S \subseteq \{1, \ldots, n\}$ of size s and an $n \times r$ matrix M from the following distribution. First, choose a set of uniformly random but distinct values $a_1, \ldots, a_n$ from $\mathbb{F}_q$. Set the ith row M_i to

$$M_i = \begin{cases} [a_i \; a_i^2 \; \cdots \; a_i^{s/3} \; 0 \; \cdots \; 0], & \text{if } i \in S, \\ [a_i \; a_i^2 \; \cdots \; a_i^{s/3} \; a_i^{s/3+1} \; \cdots \; a_i^r], & \text{if } i \notin S. \end{cases}$$

The secret key is the pair (S, M) and the public key is the matrix $P = MR$, where R is a random $r \times r$ matrix over $\mathbb{F}_q$ with determinant one. (Such a matrix can be efficiently sampled.)

Encryption: Given a public key P, to encrypt a message $m \in F_q$, choose a uniformly random $x \in \mathbb{F}_q^r$ and a noise vector $e \in \mathbb{F}_q^n$ by choosing each of its entries independently at random from $\tilde{\eta}$. Output the vector $Px + m\mathbf{1} + e$, where $\mathbf{1} \in \mathbb{F}_q^n$ is the all ones vector.

Decryption: Given a secret key (S, M), to decrypt a ciphertext $c \in \mathbb{F}_q^n$, first find a solution to the following system of $s/3 + 1$ linear equations over variables $y_i \in \mathbb{F}_q, i \in S$

$$\begin{aligned} \textstyle\sum_{i \in S} y_i M_i &= 0 \\ \textstyle\sum_{i \in S} y_i &= 1 \end{aligned} \tag{3.1}$$

with $y_i = 0$ when $i \notin S$. Output the value $\sum_{i \in [n]} y_i c_i$.

To understand the functionality of this scheme, let us first assume that no noise is present, that is $\tilde{\eta}$ always outputs zero. The decryption of an encryption of m is given by

$$y^T(Px + m\mathbf{1}) = (y^T M)Rx + m \cdot y^T \mathbf{1} = \Big(\sum\nolimits_{i \in S} y_i M_i\Big)Rx + m \sum\nolimits_{i \in S} y_i = m$$

by the constraints (3.1) imposed on y_i. We must argue that these constraints can be

simultaneously satisfied. This follows from the fact that the matrix specifying the system of equations (3.1) is an $s \times (s/3 + 1)$ Vandermonde matrix, which has full rank and is therefore left-invertible.

When noise is present in the encryption, the decryption could produce the wrong answer when at least one of the noisy elements makes it inside the hidden set S. By a union bound this happens with probability at most ηs, where $\eta = \Pr[\tilde{\eta} \neq 0]$ is the noise rate of the scheme.

3.1.2 Relation with other cryptosystems

While we are unable to argue the security of our proposed scheme by formal reduction to a previously studied one, we describe how our scheme combines ideas from the existing cryptosystems of McEliece and Applebaum, Barak, and Wigderson (ABW), with an eye towards inheriting the security features of these schemes. We take some small liberties in our discussion of these encryption schemes in order to emphasize the parallels to our proposed scheme.

In the McEliece cryptosystem based on the Reed-Solomon code, the public key looks exactly like in our scheme, except that the secret subset S is empty (i.e., $s = 0$). The syntax and semantics of the encryption, however, are somewhat different. The message set is $\mathbb{F}_q^r$ and an encryption of a message $x \in \mathbb{F}_q^r$ has the form $Px + e$, which looks like a noisy codeword of the Reed-Solomon code.[1] Decryption is performed by applying an error-correction algorithm to this codeword. What prevents the adversary from applying the error-correction himself is the fact that the (randomized) evaluation points of the Reed-Solomon code are not revealed in the public key, owing to the presence of the "key scrambling" matrix R.

In our proposed cryptosystem, the vector $x \in \mathbb{F}_q^r$ does not represent the message but is used to randomize the encryption. Since P and M are generator matrices of the same linear code, the encryption of a message $m \in \mathbb{F}_q$ can be viewed as an affine shift of a random codeword of this code by m units in every coordinate. To thwart decoding by inverting this affine transformation, a noise is injected into some of the coordinates. The ability to decrypt now relies not on the existence of efficient error-correction for the Reed-Solomon code, but on the trapdoor S. The submatrix M_S of M indexed by the rows of S has a similar structure to the whole matrix M, but on a smaller scale. The scale s of this "self-similarity" will be chosen small enough so that noise is unlikely to make it into the codeword coordinates indexed by S, allowing for very simple decoding

[1] One security issue is that these ciphertexts are not message indistinguishable.

via linear algebra.

Thus at a structural level, our proposed cryptosystem is quite similar to the ABW cryptosystem. Besides the fact that the ABW system operates over the field $\mathbb{F}_2$ while our system will be instantiated over a larger field, the main difference is in the choice of the public key matrix P. In the ABW system, the choice of this matrix is constrained by the fact that the encoding needs to be performed in a local manner. In our case, we will need M (and therefore P) to have specific algebraic structure that enables homomorphic operations.

The (private-key) proposals of Armknecht and Sadeghi [AS08] and Armknecht *et al.* [AAPS11] also bear similarity to our construction. In these schemes as in ours, the functional part of the ciphertext (i.e., the part projected onto S) can be viewed as evaluations of polynomial p that is random conditioned on the value $p(0)$, which encodes the message m. In this view, addition and multiplication of ciphertexts correspond to the same operations on the respective polynomials. By construction, these schemes support a very limited number of homomorphic operations.

3.1.3 Parameters and security

We now turn to arguing the security of our scheme against certain natural attacks. The form of security that we aim to achieve is the standard notion of (s, ε) (key independent) message indistinguishability, which asks that for every pair of messages $m, m' \in F_q$, the encryptions of m and m' are indistinguishable with advantage ε by circuits of size s that are given the public key, where the randomness is taken over the choice of keys.[2]

We describe the attacks at a somewhat informal level in order to gain intuition about the setting of parameters n, q, r, s, and η for which the proposed scheme could be secure. For convenience in further discussion, n will play the role of a security parameter and we propose values for the other parameters in terms of n. Ultimately all of these parameters will be polynomially related to n; the exact polynomial dependencies, which are chosen with some foresight, are described by a constant $\alpha > 0$, whose significance will become apparent in Section 3.4.1.

Recover the hidden subset S from the public key. A natural attack for the adversary is to locate or guess the hidden subset S. A brute-force search would go over all $\binom{n}{s}$ possible candidates for S. To obtain non-negligible security, one should choose s to increase asymptotically with n.

[2]Security can be proved even if m and m' are allowed to depend on the public key, but to avoid some technical complications in the definitions we present our results with respect to the weaker notion.

Here is a more sophisticated kind of attack that attempts to obtain information about S. A statistical way to distinguish the rows of P that are indexed by S from the other ones is based on the dimension of the hidden vectors in the matrix P. For the purposes of describing this attack we can pretend that $P = M$, as the attack only relies on the column space of P, which is identical for the two matrices. One can attempt to locate the rows in M_S by calculating the rank of various $k \times r$ submatrices D of M. If D turns out not to be of full rank, then D must contain a vector in S (for otherwise D would be a Vandermonde matrix and therefore of full rank). By performing such rank calculations one could expect to find information about the subset S.

In Appendix A we show that for any $t \times r$ submatrix D (depending on S) the rank of D is full with probability at least $1 - O(r^2/q)$, unless D contains at least $s/3 + 1 + \max\{t - r, 0\}$ rows from M_S. The probability is taken over the random choice of $a_1, \ldots, a_n$ in the key generation algorithm. A simple calculation shows that if D were chosen at random (for any choice of t), it would be rank deficient with probability at most $\min\{O(r^2/q), 1/\binom{n}{\Omega(s)}\}$.

Another type of attack our system is potentially vulnerable to is the Sidelnikov-Shestakov attack [SS92]. Wieschebrink [Wie10] shows that in the McEliece cryptosystem instantiated with generalized Reed-Solomon Codes, the secret key can be recovered from the public key in time linear n and cubic in the field size q. This attack gives a way to compute the scrambled evaluation points from the public key matrix, provided two of them are known in advance. If q is small, these two points can be guessed efficiently. One can exploit this attack to recover the matrix M and therefore reveal S.

By setting $s = n^{\alpha/4}$ and q on the order of 2^{n^α}, we ensure that these attacks require exponential time, or only give inverse exponential success probability .

Exploit the special properties of M_S in the public key. In our decryption algorithm it was crucial that the rows of the matrix M_S satisfy the constraints of the linear system (3.1). However this special structure of M_S could be potentially exploited by an adversary. For instance, an adversary may set up a system of equations analogous to (3.1), but over all indices of the ciphertext instead only of those in S. Specifically, the adversary sets up the following system of equations over variables $y_i, i \in [n]$:

$$\begin{aligned} \textstyle\sum_{i \in [n]} y_i P_i &= 0 \\ \textstyle\sum_{i \in [n]} y_i &= 1. \end{aligned}$$

Notice that the solution space of this system does not change if P is replaced by M, and so in particular it contains all the solutions to the system (3.1) (with $y_i = 0$ for $i \notin S$).

If the adversary is lucky, the solution space will contain *only* the solutions to (3.1) so by solving the system he would gain the ability to decrypt.

By choosing r to be sufficiently smaller than n—we set $r = n^{1-\alpha/8}$—we can ensure that the system set up by the adversary has abundantly many solutions, most of which will be forced to have very large hamming weight. Such solutions are useless for the decoding, as long as η is not trivially small, because the noise in the ciphertext is likely to affect some nonzero coordinates of y.

Our homomorphic algorithms rely on one additional property of the matrix M_S, namely the existence of solutions to the more constrained linear system (3.2) described in Section 3.2. We can argue that the analogous attack fails by a similar argument as to the one given here. Generally, our intuition is that we can handle attacks that exploit the similarity between the matrices M and (the nonzero part of) M_S by choosing the rows-to-columns aspect ratio of M to be substantially larger than the rows-to-columns aspect ratio of M_S, which is constant.

Recover the randomness x used in the encryption. If the noise rate η in the encryption is too small, the adversary may be able to recover x from, say, an encryption of 0. For instance, if the noise rate η is smaller than $1/r$, then in an encryption of 0 of the form $Px + e$ it would happen with constant probability that no noise makes it into the first r bits of the encryption. In that case, the adversary could recover the randomness by inverting the first r bits of the ciphertext.

We set the noise rate η to $1/n^{1-\alpha/4}$. Since $r = n^{1-\alpha/8}$, it follows that any projection of the bits of a ciphertext of linear length is likely to contain noise, which would make it exponentially hard to recover the randomness x.

Taking all these factors into consideration, we are now ready to conjecture the security of our proposed cryptosystem $\mathbf{K}$.

Conjecture 3.1. *For every $\alpha > 0$ there exists $\gamma > 0$ such that the cryptosystem $\mathbf{K}$ with parameters $r = n^{1-\alpha/8}$, $\eta = 1/n^{1-\alpha/4}$, $s = n^{\alpha/4}$ and $q \geq 2^{n^\alpha}$ is $(2^{n^\gamma}, 2^{-n^\gamma})$-message indistinguishable, for all n that are sufficiently large.*

Conjecture 3.1 follows from the possibly stronger pseudorandomness assumption that the distribution $(PK, \mathbf{Enc}_{PK}(0))$ is $(2^{n^\gamma}, 2^{-n^\gamma})$-computationally indistinguishable from (PK, U), where PK is the public key, $\mathbf{Enc}_{PK}(0)$ is a random encryption of 0 under PK, and U is a uniformly random string of length n independent of PK.

We will use $\mathbf{K}_q(n)$ to denote an instantiation of the cryptosystem $\mathbf{K}$ with the parameters from Conjecture 3.1 (except for q which we leave as a free parameter).

3.1.4 Our main result

For technical simplicity we state our definitions and results in the non-uniform setting (i.e. all components are described as circuits instead of algorithms). An extension to the uniform setting is straightforward.

In our definition of homomorphic encryption we wish to distinguish between the standard decryption algorithm, which applies to encryptions of bits, and the homomorphic decryption algorithm, which applies to the output of the homomorphic evaluation circuit. Also, unlike previous homomorphic encryption schemes, ours carries the risk of a setup error, which we account for in the definition.

Owing to this risk of error, it is possible that some of the inputs provided to the homomorphic evaluation circuit are themselves corrupted. To provide for this possibility, we give a somewhat more general definition of homomorphic evaluation: Instead of requiring that the circuit works well on *encryptions* of the inputs (which are not even well-defined in the setting of error-prone probabilistic encryption), we ask that they work on inputs that *decrypt* to the correct value. This feature of the definition will be very useful in the proofs.

Definition 3.2. A *homomorphic encryption scheme* with setup error κ for circuit class $\mathcal{C} = \{C\colon B^m \to B\}$ (where B is a subset of the message set) consists of five circuits $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec}, \mathbf{Eval}, \mathbf{HDec})$, where $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ is a (probabilistic) public-key encryption scheme (for a formal definition see e.g. [Gol04]), and **Eval** and **HDec** are (deterministic) circuits that satisfy

$$\Pr[\mathbf{HDec}_{SK}(\mathbf{Eval}_{PK}(C, c_1, \ldots, c_m)) = C(m_1, \ldots, m_m)] \geq 1 - \kappa$$

for every circuit $C \in \mathcal{C}$, every message $m \in \{0,1\}^m$, and every collection of ciphertexts $c_1, \ldots, c_m$ such that $\mathbf{Dec}_{SK}(c_i) = m_i$ for every i. The probability is taken over the choice of keys $(SK, PK) \sim \mathbf{Gen}$.

Let $C\colon \{0,1\}^m \to \{0,1\}$ be a boolean circuit with binary addition (i.e. XOR) and multiplication (i.e. AND) gates of fan-in two. The *depth* of C is the maximum number of gates on a directed path of C. We let $\mathcal{C}_{cs,d}$ denote the class of such circuits with circuit size cs and depth d.

Our main result is a construction of a "layered" homomorphic encryption scheme **HOM** based on **K**, which is fully described in Section 3.5. The following theorem summarizes the functionality and security properties of our scheme. The parameter k controls the setup error and can be instantiated to any desired value.

Theorem 3.3. *Let $q \leq 2^n$ be a power of two. Assume that the public-key encryption $\mathbf{K}_q(n)$ is $(s(n), \varepsilon(n))$-message indistinguishable for every n (where $s(n)$ and $1/\varepsilon(n)$ are nondecreasing functions of n). Then* **HOM** *is a $(s(n^{0.1}) - dk \cdot \mathrm{poly}(n), O(dkn^{1.8}\varepsilon(n^{0.1})))$-message indistinguishable homomorphic encryption scheme for $\mathcal{C}_{cs,d}$ with key length at most $O(dkn)$, encryption length $O(kn)$, encryption error $2^{-\Omega(k)}$, and setup error $d \cdot 2^{-\Omega(k)}$.*

3.1.5 Overview of HOM

To begin, in Section 3.2 we show that the operations of *pointwise* addition and multiplication already enjoy certain somewhat homomorphic properties, which are sufficient to handle one layer of homomorphic multiplications. We formalize these properties using the new notion of *encryption spaces*, which may be a convenient conceptual tool for studying the functionality of homomorphic encryptions. The analysis relies on the special structure of the matrix M, specifically on the large redundancy of the constraint system (3.2).

In Section 3.3 we give a formal definition of recryption, a notion crucial in our and other constructions. We show that somewhat homomorphic operations together with secure recryption gives secure homomorphic schemes. We apply an idea of Gentry to obtain a recryption for our public-key scheme **K**. Unfortunately, owing to the inherent noise in our encryptions, the recryption substantially increases the length of ciphertexts, and the resulting homomorphic scheme has a noticeable setup error.

Section 3.4 contains the main technical contributions of our work which address these deficiencies. We first give a secure length-preserving recryption based on a recursive application of the length-increasing recryption from Section 3.3 which we use to obtain homomorphic noise correction. We then give a generic mechanism for reducing the setup error, which extends von Neumann's method of building reliable circuits from unreliable components [vN56] to the homomorphic setting.

Combining these results, we give the construction of **HOM** and prove Theorem 3.3 in Section 3.5.

3.2 Encryption spaces and somewhat homomorphic operations

Since homomorphism of encryptions is a functionality rather than a security requirement, we feel that it is useful to decouple the functionality and security properties of the

schemes under discussion. For this purpose we introduce the notion of an *encryption space* which is concerned with the set-theoretic properties of encryptions and abstracts away their statistical properties.

Definition 3.4. An *encryption space* over message set Σ and ciphertext set Ξ is a triple $(Keys, Enc, Dec)$, where

- $Keys$ is a set of admissible key pairs (PK, SK),
- $Enc_{PK}(\cdot)$ is a function that maps messages $m \in \Sigma$ into subsets of valid encyptions $Enc_{PK}(m) \subseteq \Xi$, and
- $Dec_{SK}(\cdot)$ is a function that maps messages $m \in \Sigma$ into mutually disjoint decryptable ciphertexts $Dec_{SK}(m) \subseteq \Xi$.

with the property that $Enc_{PK}(m) \subseteq Dec_{SK}(m)$ for every $(PK, SK) \in Keys$ and $m \in \Sigma$.

We will say that a public-key encryption scheme $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ *implements* the encryption space $(Keys, Enc, Dec)$ with encryption error δ if (1) The support of the output distribution of **Gen** is contained in $Keys$; (2) For every m and PK, $\Pr[\mathbf{Enc}_{PK}(m) \in Enc_{PK}(m)] \geq 1 - \delta$; and (3) For every SK and $c \in Dec_{SK}(m)$, $\mathbf{Dec}_{SK}(c) = m$.

An encryption space for K Notice that for the functionality of the scheme **K**, it only matters what happens to the part of the ciphertext that falls inside the hidden subset S. Our definition of the encryption space $K = (Keys, Enc, Dec)$ for **K** will capture this intuition. However, we will equip K with an additional property which will be crucial to achieve somewhat homomorphic encryption.

We set $Keys$ to be the support of the key generation algorithm **Gen** and $Enc_{PK}(m)$ to be the set of all ciphertexts that take value $Mx + m\mathbf{1} + f$, where $f_i = 0$ when $i \in S$ and f_i can be arbitrary when $i \notin S$. We define $Dec_{SK}(m)$ as the collection of all ciphertexts c that satisfy $y^T c = m$ for some arbitrary but fixed y that solves the following system of linear equations:

$$\begin{aligned} \textstyle\sum_{i \in S} y_i (M_i \otimes M_i) &= 0 \\ \textstyle\sum_{i \in S} y_i M_i &= 0 \\ \textstyle\sum_{i \in S} y_i &= 1 \end{aligned} \tag{3.2}$$

with $y_i = 0$ when $i \notin S$. Here $M_i \otimes M_i$ denotes the tensor product of M_i with itself, which we view as an s^2-dimensional vector (after removing the zero entries) whose (j,k)th entry is $a_i^j a_i^k = a_i^{j+k}$. Notice that the system (3.2) is more constrained than the system (3.1) as it includes additional equations. These equations will play a crucial role in enabling homomorphic multiplication.

Claim 3.5. *K is an encryption space over message set $\mathbb{F}_q$.*

Proof. To make sense of the definition of K we must first argue that the system (3.2) has at least one solution y. Here is where the structure of the Reed-Solomon code comes in handy: Although the system (3.2) has as many as s^2 equations, they all repeat the following set of $2s/3+1$ equations:

$$\begin{aligned} \textstyle\sum_{i\in S} y_i a_i^k &= 0 \quad \text{for } k = 1, 2, \ldots, 2s/3 \\ \textstyle\sum_{i\in S} y_i &= 1. \end{aligned}$$

The matrix of this system is an $s \times (2s/3+1)$ Vandermonde matrix and is therefore left-invertible, so the system is guaranteed to have a solution.

The disjointness of the sets $Dec_{SK}(m)$ is immediate. We now show that $Enc_{PK}(m) \subseteq Dec_{SK}(m)$ for every $m \in \mathbb{F}_q$. Let c be of the form $Mx+m\mathbf{1}+f$ and let y be any solution to (3.2). Since $y^T f = 0$, we have that

$$y^T c = y^T(Mx + m\mathbf{1}) = \Big(\textstyle\sum_{i\in S} y_i M_i\Big)x + m\Big(\sum_{i\in S} y_i\Big) = m$$

which proves the claim. □

The next fact follows directly from the definitions of $\mathbf{K}$ and K.

Fact 3.6. *The encryption scheme $\mathbf{K}$ implements the encryption space K with encryption error ηs.*

Somewhat homomorphic operations We now define the notion of homomorphic and somewhat homomorphic operations on ciphertexts, which plays an important role in homomorphic constructions.

Definition 3.7. Let $(Keys, Enc, Dec)$ be an encryption space with message set Σ and ciphertext set Ξ. Let $\circ$ and $\circledcirc$ be binary operations on Σ and Ξ, respectively.

- We will say $\circledcirc$ is *homomorphic* for $\circ$ if for every $(PK, SK) \in Keys$ and $m, m' \in \mathbb{F}_q$,

$$Enc_{PK}(m) \circledcirc Enc_{PK}(m') \subseteq Enc_{PK}(m \circ m').$$

- We will say $\circledcirc$ is *somewhat homomorphic* for $\circ$ if for every $(PK, SK) \in Keys$ and $m, m' \in \mathbb{F}_q$,

$$Enc_{PK}(m) \circledcirc Enc_{PK}(m') \subseteq Dec_{SK}(m \circ m').$$

Here, $\circledcirc$ is extended to an operation on sets in the natural way. The definitions extend naturally to unary operations. Now let $\oplus$ and $\odot$ denote pointwise addition and pointwise multiplication over $\mathbb{F}_q^n$ respectively, and let $\gamma\cdot$ denote multiplication of a vector in $\mathbb{F}_q^n$ by the fixed scalar γ.

Claim 3.8. *With respect to the encryption space K, $\oplus$ is homomorphic for addition, $\gamma\cdot$ is homomorphic for multiplication by the scalar γ, and $\odot$ is somewhat homomorphic for multiplication.*

Proof. Let $c = Mx + m\mathbf{1} + f$ and $c' = Mx' + m'\mathbf{1} + f'$, where $f_i = f'_i = 0$ when $i \in S$. Then $c \oplus c' = M(x + x') + (m + m')\mathbf{1} + (f + f')$, which is in $Enc_{PK}(m + m')$, proving homomorphism for additions. Scalar multiplications are similar. For multiplications, let y be any solution to (3.2) and notice that

$$\begin{aligned} y^T(c \odot c') &= \sum\nolimits_{i=1}^{n} y_i(Mx + m\mathbf{1} + f)_i(Mx' + m'\mathbf{1} + f')_i \\ &= \sum\nolimits_{i\in S} y_i(Mx + m\mathbf{1})_i(Mx' + m'\mathbf{1})_i \\ &= \sum\nolimits_{i\in S} y_i(M_i \otimes M_i)^T(x \otimes x') + m \cdot y^T Mx' + m' \cdot y^T Mx + mm' \cdot y^T\mathbf{1} \\ &= mm' \end{aligned}$$

since by the constraints (3.2) we have $\sum_{i\in S} y_i(M_i \otimes M_i) = 0$, $y^T M = 0$, and $y^T\mathbf{1} = 1$. □

Claim 3.8 already enables homomorphic evaluation under $\mathbf{K}$ of circuits that have at most one layer of multiplication gates. To do more, we need a homomorphic way of turning ciphertexts of the form $Dec_{SK}(m)$ into ciphertexts of the form $Enc_{PK}(m)$. While we will not achieve this—at least not under the desired security assumption—in the following sections we will show how to convert $Dec_{SK}(m)$ into $Enc_{PK'}(m)$, where PK' is a different public key. We describe this process of *recryption* in the following section.

3.3 Recryption

We now define the functionality and security requirements of recryption. We then prove a composition theorem which shows how to obtain homomorphic encryption from recryption and a basis of somewhat homomorphic operations.

Intuitively, a recryption circuit takes a decryption under keys (PK, SK) and outputs an encryption under keys (PK', SK'). To do this the circuit will access some auxiliary information about the secret key SK which will be "hidden" under PK'. We model

this auxiliary information by an *auxiliary key information* function $I(SK, PK')$. One complication that occurs in our instantiations of recryption is that the function I will be randomized, and we will have to account for the possibility that it produces incorrect information about the key pair.

Definition 3.9. Let $E = (Keys, Enc, Dec)$ and $E' = (Keys', Enc', Dec')$ be encryption spaces over the same message set. A (deterministic) circuit $\mathbf{ReEnc}_{I(\cdot)}(\cdot)$ is a *recryption* from E to E' with auxiliary key information I and key error κ if for every admissible pair $(PK, SK) \in Keys, (PK', SK') \in Keys'$,

$$\Pr_I[\mathbf{ReEnc}_{I(SK,PK')}(c) \in Enc_{PK'}(m) \text{ for every message } m \text{ and every } c \in Dec_{SK}(m)] \geq 1 - \kappa$$

where the outer probability is taken only over the randomness of I.

To define security, let $\mathbf{E}$ and $\mathbf{E}'$ be encryption schemes that implement E and E' respectively. We will say **ReEnc** is $(s \to s', \varepsilon \to \varepsilon')$-secure provided that for every pair of messages m_1 and m_2, if $(PK, \mathbf{Enc}_{PK}(m_1))$ and $(PK, \mathbf{Enc}_{PK}(m_2))$ are (s, ε) indistinguishable, then $(PK, PK', I(SK, PK'), \mathbf{Enc}_{PK}(m_1))$ and $(PK, PK', I(SK, PK'), \mathbf{Enc}_{PK}(m_2))$ are (s', ε') indistinguishable.

We now show how to combine somewhat homomorphic operations and recryption in order to obtain homomorphic encryption. One small complication is that in our definition of recryption we allow that the two schemes $\mathbf{E}$ and $\mathbf{E}'$ are different. This is an important feature that will help us achieve the definition initially. So when we apply d levels of recryption, we will work with a chain of public-key encryption schemes $\mathbf{E}_0, \ldots, \mathbf{E}_d$.

Let $\mathbf{E}_0, \ldots, \mathbf{E}_d$ be public-key encryption schemes so that $\mathbf{E}_i$ implements encryption space E_i. Assume $\mathbf{ReEnc}_i$ is a recryption from E_i to E_{i+1} with auxiliary information I_i.

Let C be a circuit with binary gates, each of which has a homomorphic or somewhat homomorphic implementation in all of the spaces E_i. Abusing terminology, we will call these gates homomorphic and somewhat homomorphic gates, respectively. The *somewhat homomorphic depth* of C is the largest number of somewhat homomorphic gates on any directed path in any circuit in $\mathcal{C}$. Without loss of generality (by adding some dummy gates), we will assume that the somewhat homomorphic gates in C are layered, i.e. every path in every circuit has exactly the same number of somewhat homomorphic gates. Let $\mathcal{C}^{\circ}_{cs,d}$ be the class of circuits of size cs and somewhat homomorphic depth d.

Homomorphic template $\mathbf{T}(\mathbf{E}_0, \ldots, \mathbf{E}_d)$ for $\mathcal{C}^{\circ}_{cs,d}$

Key generation: Generate key pairs (PK_i, SK_i) uniformly at random for every i. Generate auxiliary key information $I_i(SK_i, PK_{i+1})$ uniformly at random for every i. The secret key is (SK_0, SK_d). The public key is $(PK_0, \ldots, PK_d, I_0, \ldots, I_{d-1})$.

Encryption and decryption are the same as in $\mathbf{E}_0$ using the key pair (PK_0, SK_0).

Homomorphic decryption is the same as in $\mathbf{E}_d$ using the secret key SK_d.

Homomorphic evaluation: Given a layered circuit C, replace every homomorphic gate $+$ of C by its homomorphic implementation $\oplus$. At every somewhat homomorphic layer i, replace the somewhat homomorphic gates $\cdot$ by their somewhat homomorphic implementations $\odot$ followed by $\mathbf{ReEnc}_i$. Add recryption gates $\mathbf{ReEnc}_0$ to the input level. Perform the evaluations of the ciphertext, using auxiliary information I_i for $\mathbf{ReEnc}_i$. Output the resulting ciphertext.

The following two statements capture the functionality and security properties of this scheme; we omit the easy proofs.

Proposition 3.10. *Suppose* $\mathbf{ReEnc}_i$ *has key error at most* κ*. Then* $\mathbf{T}(\mathbf{E}_0, \ldots, \mathbf{E}_d)$ *is a homomorphic encryption scheme with setup error at most* $d \cdot \kappa$*.*

Claim 3.11. *Suppose* $\mathbf{E}_0$ *is* (s_0, ε_0)*-message indistinguishable and* $ReEnc_i$ *is* $(s_i \to s_{i+1}, \varepsilon_i \to \varepsilon_{i+1})$ *secure for every* i*. Then* $\mathbf{T}(\mathbf{E}_0, \ldots, \mathbf{E}_d)$ *is* (s_d, ε_d)*-message indistinguishable.*

3.3.1 Constructing recryption

We now give a construction of a recryption from the family of encryptions $\mathbf{K}_q(n)$. Let $\mathbf{K}_q(n)$ and $\mathbf{K}_q(n')$ be two instantiations of $\mathbf{K}$ with a different hardness parameter, specifically with $n' > n$. To simplify notation we will identify the two encryption schemes with their corresponding encryption spaces.

Our construction of a recryption from $\mathbf{K}_q(n)$ to $\mathbf{K}_q(n')$ is based on Gentry's ingenious idea of homomorphically evaluating the decryption circuit of $\mathbf{K}_q(n)$. The decryption circuit in our scheme is extremely simple as it only uses homomorphic additions. However, one important complication in our scheme is the possibility of encryption errors. While for a single encryption the likelihood of an error occurring is small, when we apply the encryption to all the coordinates of the "secret key" the error becomes substantial. Our choice of parameters for $\mathbf{K}_q(\cdot)$ is essential for controlling the error; it will allow us to

tolerate a substantial amount of error provided we choose n' to be large enough in terms of n.

We now describe the recryption. Let y be the designated solution to the system (3.2), which specifies the decryption space of $\mathbf{K}_q(n)$. Recall that $y_i = 0$ whenever i is outside the hidden subset S. The auxiliary key information $I(SK, PK')$ consists of the encryptions $z_1 = \mathbf{Enc}_{PK'}(y_1), \ldots, z_n = \mathbf{Enc}_{PK'}(y_n)$, where all encryptions are performed independently. Each of these encryptions is a vector in $\mathbb{F}_q^{n'}$. The recryption is given by

$$\mathbf{ReEnc}_{z_1,\ldots,z_n}(c) = c_1 z_1 + \cdots + c_n z_n.$$

Claim 3.12. **ReEnc** *is a recryption from* $\mathbf{K}_q(n)$ *to* $\mathbf{K}_q(n^{1+\alpha})$ *with auxiliary information* I *and key error* $n^{-\alpha(1-\alpha)/2}$.

Proof. Recall that z_i has the form $M'x_i + y_i\mathbf{1} + e_i$, where e_i is an error vector with error rate η'. We will say the output of $I(PK', SK)$ is *good* if for all $i \in [n]$, all the entries of e_i that fall inside the hidden subset S' are zero. By a union bound, the probability that $I(PK', SK)$ is not good is at most

$$\eta' s' n = n^{-(1+\alpha)(1-\alpha/4)} \cdot n^{(1+\alpha)(\alpha/4)} \cdot n = n^{-\alpha(1-\alpha)/2}.$$

We now show that if $I(PK', SK)$ is good then $\mathbf{ReEnc}_I(c) \in Enc_{PK'}(m)$ for every $c \in Dec_{SK}(m)$. Recall that $Enc_{PK'}(m)$ contains those ciphertexts that take value $M'_{S'}x + m\mathbf{1}$ inside S' (for some x) and can take arbitrary value outside S'. Since I is good, we know that the projection of z_i onto S' has the form $M'_{S'}x_i + y_i\mathbf{1}$. Therefore the projection of $\mathbf{ReEnc}_I(c)$ to S' has the form

$$\sum_{i=1}^{n} c_i(M'_{S'}x_i + y_i\mathbf{1}) = M'_{S'}x + (c^T y)\mathbf{1} = M'_{S'}x + m\mathbf{1}$$

where $x = \sum c_i x_i$. □

The following security claim can be derived by a hybrid argument.

Claim 3.13. *If* $\mathbf{K}_q(n')$ *is* (s, ε')*-message indistinguishable then* **ReEnc** *is* $(s \to s - \mathrm{poly}(n), \varepsilon \to \varepsilon + n\varepsilon')$*-secure.*

Assume $\mathbf{K}_q(n)$ is $(s, \varepsilon(n))$-message indistinguishable for every n, where $\varepsilon(n)$ is non-increasing. Instantiating the template $\mathbf{T}(\mathbf{E}_0, \ldots, \mathbf{E}_d)$ with the encryption schemes $\mathbf{E}_i = \mathbf{K}_q(n^{(1+\alpha)^i})$, we obtain a family of homomorphic encryption schemes $\mathbf{BASIC}(n)$ for circuits $C\colon \mathbb{F}_q^m \to \mathbb{F}_q$ with addition, scalar multiplication, and binary multiplication gates

of size cs and multiplication depth d with key length and encryption length $O(n^{(1+\alpha)^d})$ and setup error $dn^{-\alpha(1-\alpha)/2}$ that are $(s - d \cdot \mathrm{poly}(n), O(n^{(1+\alpha)^{d-1}}\varepsilon(n)))$-message indistinguishable.

3.4 Optimizing recryption

We now describe two transformations to recryption. The purpose of the first transformation is to eliminate the blowup in the security parameter in Claim 3.12. The second one is a generic technique for reducing the key error.

3.4.1 Improving the key length

Let us revisit the homomorphic scheme **BASIC** from the previous section. For convenience we will introduce a change of parameters. After performing d layers of homomorphic multiplication, the length of the ciphertext went from n_0 to $n = n_0^{(1+\alpha)^d}$. We will describe a recryption from $\mathbf{K}_q(n)$ to $\mathbf{K}_q(n)$.

What we would like to do is use the transformation from Claim 3.12, but without increasing the length n. As we noted, this is difficult to do owing to the large amount of encryption error that accumulates into the auxiliary key information. Now let us attempt to *reduce* the recryption length by moving from $\mathbf{K}_q(n)$ to $\mathbf{K}_q(n_0)$. This appears even less reasonable, as $\mathbf{K}_q(n_0)$ has even greater encryption error than $\mathbf{K}_q(n)$. But one advantage of working with $\mathbf{K}_q(n_0)$ is that the scheme **BASIC** already allows us to do homomorphic evaluation over its ciphertexts. Our idea is to apply **BASIC** to a "correction circuit" $CORR$ whose purpose is to eliminate the encryption errors introduced when encrypting the secret key information about $\mathbf{K}_q(n)$ using $\mathbf{K}_q(n_0)$.

To carry out this idea, we have to be somewhat careful about the design of $CORR$. Here, the value of the parameter α will play an important role. If $CORR$ is too deep the security suffers, as it is dictated by n_0, while the encryption length is $n \gg n_0$. For a careful choice of the parameters, we can ensure that $CORR$ has constant depth, which will enable us to produce length-preserving recryptions of size n with security parameter polynomial in n.

We will assume that q is a power of two. Let d be an even constant (we later set it to 8). Let (PK, SK) and (PK', SK') be two admissible key pairs for $\mathbf{K}_q(n)$.

Recryption. We generate the auxiliary key information as follows. First, sample a sequence of independent key pairs $(PK_0, SK_0), \ldots, (PK_{d-1}, SK_{d-1})$, where (PK_i, SK_i) comes from $\mathbf{Gen}(n_0^{(1+\alpha)^i})$. Let $y \in \mathbb{F}_q^n$ specify the decryption space of $\mathbf{K}_q(n)$. The

auxiliary information is generated as follows. Let γ be a generator for the field extension $\mathbb{F}_q$ over $\mathbb{F}_2$.

1. **Encrypt:** For each coordinate y_i of y, expand as $y_i = y_{i0}+\gamma y_{i1}+\ldots+\gamma^{\log q-1}y_{i\log q-1}$ with $y_{ij} \in \{0,1\}$. For every i, j, create 2^d independent ciphertexts $c_{ij}^k = \mathbf{Enc}_{PK_0}(y_{ij})$, where k ranges from 1 to 2^d.

2. **Correct:** For every i, j, calculate $z_{ij} = \mathbf{Eval}(CORR, c_{ij}^1, \ldots, c_{ij}^{2^d})$, where **Eval** is the evaluation algorithm for **BASIC** when the key generation algorithm is instantiated with the keys $(PK_0, SK_0), \ldots, (PK_{d-1}, SK_{d-1}), (PK', SK')$, and $CORR\colon \{0,1\}^{2^d} \to \{0,1\}$ is the circuit described below.

3. **Output:** Let $z_i = z_{i0}+\gamma z_{i1}+\ldots+\gamma^{\log q-1}z_{i\log q-1}$. Output the vector $I(SK, PK') = (z_1, \ldots, z_n)$.

As before, the recryption procedure is $\mathbf{ReEnc}_{z_1,\ldots,z_n}(c) = c_1z_1 + \cdots + c_nz_n$.

We now describe the correction circuit. The purpose of this circuit is to eliminate the errors accumulated in the encryption, which suggests using majority. However we also need to have fine control over the depth of the circuit. Since the errors of various encryptions are independent, it is natural to use a recursive majority-type construction in order to correct the error from one layer to the next. For our analysis, it will be convenient to make $CORR$ be a full binary tree of depth d where d is even and all the gates are of the type $G(x, y) = 1 - xy$. When restricted over $\{0,1\}$ inputs, this is a NAND tree.

Proposition 3.14. *For $\alpha \le 1/4$ and $d = 8$,* **ReEnc** *is a recryption from $\mathbf{K}_q(n)$ to $\mathbf{K}_q(n)$ with auxiliary key information I and key error $O(n^{-0.5})$.*

Proof. With probability $dn^{-\alpha(1-\alpha)/2}$ over the choice of keys, we know that the circuit **Eval** makes no mistake on its input. Let us assume this is the case.

We will show that with probability $1 - O(n^{-0.5})$, $z_{ij} \in Enc_{PK'}(y_{ij})$ for every pair (i, j). By the homomorphic property of additions and scalar multiplications, it follows that $z_i \in Enc_{PK'}(y_i)$ for all i. The correctness of recryption then follows by the same argument as in Claim 3.12.

We fix i and j and for notational convenience we write $y = y_{ij}$, $z = z_{ij}$, $c^k = c_{ij}^k$. Let $\hat{y}^k$ denote the unique value in $\mathbb{F}_q$ such that $\mathbf{Dec}_{SK_0}(c^k) = \hat{y}^k$. Since the encryption of the y_{ij}s was performed at error rate η_0, it follows that independently for each y, $\hat{y}^k = y$ with probability $1 - \eta_0$, and otherwise $\hat{y}^k$ could be an arbitrary element in $\mathbb{F}_q$.

Let us start with the special case $d = 2$. We will argue that the $\Pr[z \notin Enc_{PK'}(y)] \leq 6\eta_0^2$. This follows from the design of the circuit $CORR$. If $CORR$ is given four inputs, three of which have the same value 0 or 1, its output will also have the same value. Therefore the event $z \notin Enc_{PK'}(y)$ can only happen if $\hat{y}^k \neq y$ for at least two values of k, which happens with probability at most $6\eta_0^2$.

By induction on (even values of) d, it follows that in general the event $z \notin Enc_{PK'}(y)$ can happen with probability at most $6^{2^{d/2}-1}\eta_0^{2^{d/2}}$. We now take a union bound over all pairs i and j and conclude that the recryption is correct with probability at least $n(\log q)(6\eta_0)^{2^{d/2}}$.

Now recall that $\log q \leq n$ and $n = n_0^{(1+\alpha)^d}$, which gives an error of

$$n_0^{2(1+\alpha)^d}(6\eta_0)^{2^{d/2}} = \frac{6^{2^{d/2}}}{n_0^{(1-\alpha/4)2^{d/2}-2(1+\alpha)^d}} \leq \frac{6^{2^{d/2}}}{n_0^{(15/16)\cdot 2^{d/2}-2\cdot(5/4)^d}} = O(n_0^{-3.07}) = O(n^{-0.5})$$

for $d = 8$ and $\alpha \leq 1/4$. □

The following claim follows by a standard hybrid argument and we omit the proof.

Claim 3.15. *Fix $\alpha \leq 1/4$ and $d = 8$ and assume $\mathbf{K}_q(n)$ is $(s(n), \varepsilon(n))$-message indistinguishable for every n, where $\varepsilon(n)$ is nonincreasing. Then for every ε_0,* **ReEnc** *is $(s(n) \to s(n^{0.1}) - \mathrm{poly}(n), \varepsilon_0 \to \varepsilon_0 + O(n^{1.8} \cdot \varepsilon(n^{0.1}))$-secure.*

3.4.2 Reducing the key error

The final optimization we perform concerns the key error of recryption. The key error of the recryption **ReEnc** from the previous section cannot be reduced beyond $1/n$. In the homomorphic template in Section 3.3, the setup error increases linearly with the number of recryptions, so we cannot apply this scheme to circuits of depth larger than n. We now introduce a generic technique for reducing this error.

Suppose we are given a recryption **ReEnc** with key error $\kappa \leq 1/32$. If we apply **ReEnc** k times in parallel to the same ciphertext but using independent instantiations of the auxiliary key information, by large deviation bounds we can expect that with probability $1 - 2^{-\Omega(k)}$, a significant majority—say a 15/16 fraction—of the recryptions will be correct. However, reapplying recryption over and over again will quickly yield overwhelming error. This calls for a boosting tool of the following kind: Given k ciphertexts out of which, say, 15/16 represent the same value, output k ciphertexts out of which a larger majority, say 31/32, now represent that value. We implement this

functionality in a circuit that we call **Boost**. For later convenience we reencrypt the outputs of **Boost**.

Definition 3.16. Let E and E' be two encryption spaces over the same message set and (PK, SK), (PK', SK') be a pair of admissible keys from the respective spaces. A *booster* of length k from E to E' with auxiliary key information $I(SK, PK')$ and key error κ is a circuit **Boost** with the following property. For every message $m \in \{0,1\}$ and ciphertexts $c_1, \ldots, c_k$ out of which at least $15k/16$ belong to $Dec_{SK}(m)$, $\mathbf{Boost}_{I(SK,PK')}(c_1, \ldots, c_k)$ outputs ciphertexts $c'_1, \ldots, c'_k$ out of which at least $31k/32$ belong to $Enc_{PK'}(m)$.

We emphasize that we only require the definition holds for messages $m \in \{0,1\}$, and not arbitrary messages in $\mathbb{F}_q$. The security definition for boosters is identical to the one for recryptions.

Our construction of boosters is based on von Neumann's idea of robust evaluation of circuits with faulty gates [vN56]. Let G be a bipartite expander graph with k vertices on each side. The circuit **Boost** will apply G to its inputs and perform a homomorphic majority at each output. Computing each of these homomorphic majorities may require some recryptions. The auxiliary key information in each of these recryptions will be independent, ensuring that with very high probability few errors will be introduced in the recryption.

The construction Assume **E** is an encryption scheme equipped with $\oplus$, $\odot$ and recryption **ReEnc** over ciphertexts of length n. Let G be an $(n, b, \lambda = 1/32)$ spectral expander [HLW06] for a sufficiently large constant b, and let $APXMAJ_b \colon \mathbb{F}_q^b \to \mathbb{F}_q$ be a circuit of depth that depends only on b (not on q) so that

$$APXMAJ_b(x_1, \ldots, x_b) = \begin{cases} 0, & \text{if at least } 7b/8 \text{ of the inputs are } 0, \\ 1, & \text{if at least } 7b/8 \text{ of the inputs are } 1. \end{cases} \tag{3.3}$$

In Appendix B we show the existence of such a circuit of size $O(b^2)$ and depth $b' = O(\log b)$.

Auxiliary key information $I(SK, PK')$: Repeat the following independently b' times, once for every output j of **Boost**: First, generate a sequence of keys $(PK_1^j, SK_1^j), \ldots, (PK_{b'-1}^j, SK_{b'-1}^j)$ and set $SK = SK_0^j, PK' = PK_{b'}^j$. Output $I'(SK_i^j, PK_{i+1}^j)$ for every i and j, where I' is the auxiliary key information for **ReEnc**.

The circuit Boost: Suppose that output j of G is connected to inputs $j_1, \ldots, j_b$. For every output j, apply the homomorphic evaluation to the circuit $APXMAJ_b$ on inputs

$c_{j_1}, \ldots, c_{j_b}$ as described in Section 3.3, but using the auxiliary key information with superscript j, and with an extra round of recryptions at the output.

Proposition 3.17. *Assume* **ReEnc** *is a recryption whose key error κ is a sufficiently small absolute constant (independent of n). Then* **Boost** *is a booster with key error $2^{-\Omega(k)}$.*

Proof. By Proposition 3.10, each of the homomorphic majority circuits has setup error at most $O(\kappa \log b)$. Since these setup errors are independent, by Chernoff bounds the chances that more than $k/64$ is at most $2^{-\Omega(k)}$. Let us assume this is not the case.

Now let B be the set of inputs of G whose value is different from $m \in \{0,1\}$. By assumption, $|B| \leq k/16$. Let S be the set of outputs of G that connect to more than $b/8$ inputs inside B. Then there are at least $|S|b/8$ edges between S and B. By the expander mixing lemma, $|S|/8k \leq |S|/16k + \lambda\sqrt{|S|/16k}$, from where $|S| \leq 16\lambda^2 k \leq k/64$ by our choice of λ.

It follows that at most $k/64 + k/64 = k/32$ outputs of **Boost** will decrypt incorrectly with probability at most $1 - 2^{-\Omega(k)}$. □

We now state the security of this construction.

Claim 3.18. *If* **ReEnc** *is $(s \to s', \varepsilon_0 \to \varepsilon_0 + \varepsilon)$-secure, then* **Boost** *is $(s \to s' - k \cdot \mathrm{poly}(n), \varepsilon_0 \to \varepsilon_0 + O(k\varepsilon))$-secure.*

3.5 The scheme HOM

To obtain our scheme **HOM**, we will apply the homomorphic template of Section 3.3 to k parallel copies of the base scheme $\mathbf{K}_q(n)$, using the booster from Section 3.4.2 to perform recryptions. Let n denote the security parameter.

Let $\mathbf{K}_q^k(n)$ denote the following scheme over message set $\mathbb{F}_q$ and ciphertext set $\mathbb{F}_q^{kn}$. The key generation algorithm is the same as in $\mathbf{K}_q(n)$. To encrypt a message m, we output k independent encryptions of m in $\mathbf{K}_q(n)$. To decrypt a ciphertext $c_1 \ldots c_k$, we apply the decryption of $\mathbf{K}_q(n)$ on each c_i and output the most frequent answer.

Let $K = (Keys, Enc, Dec)$ denote the encryption space for $\mathbf{K}_q(n)$ from Section 3.2. We now define an encryption space $K^k = (Keys, Enc^k, Dec^k)$ for $\mathbf{K}_q^k(n)$. We let $Enc^k_{PK}(m)$ consists of those ciphertexts $c_1 \ldots c_k$ for which $c_i \in Enc_{PK}(m)$ for at least $31k/32$ values of i. We let $Dec^k_{SK}(m)$ consists of those ciphertexts $c_1 \ldots c_k$ for which $c_i \in Dec_{SK}(m)$ for at least $15k/16$ values of i.

It is easy to see that if K is an encryption space for $\mathbf{K}_q(n)$ with encryption error $1/64$, then K^k is an encryption space for $\mathbf{K}_q^k(n)$ with encryption error $2^{-\Omega(k)}$. The error follows from a large deviation bound.

It is also easy to see that pointwise addition $\oplus$ and pointwise multiplication $\odot$ are somewhat homomorphic over message set $\{0,1\}$ with respect to K^k. Notice that although $\oplus$ was homomorphic for K, it is merely somewhat homomorphic for K^k, owing to the possibility of erroneous encryptions in Enc^k.

Finally, notice that the booster **Boost** from Section 3.4.2 (instantiated with the length-preserving recryption **ReEnc** from Section 3.4.1) is a *recryption* for K^k. Now define

$$\mathbf{HOM} = \mathbf{T}(\mathbf{K}_q^k(n), \ldots, \mathbf{K}_q^k(n)) \text{ with recryption } \mathbf{Boost}$$

where $\mathbf{T}$ is the homomorphic template from Section 3.3. The following two claims prove Theorem 3.3.

Claim 3.19. *The scheme* **HOM** *is a homomorphic encryption scheme for $\mathcal{C}_{cs,d}$ with key length $O(dkn)$ and setup error $d \cdot 2^{-\Omega(k)}$.*

This claim follows directly froms Proposition 3.10 and Proposition 3.17.

Claim 3.20. *Assume $\mathbf{K}_q(n)$ (with $\alpha \leq 1/4$) is $(s(n), \varepsilon(n))$-message indistinguishable, where $s(n)$ and $1/\varepsilon(n)$ are nondecreasing. Then* **HOM** *is $(s(n^{0.1}) - dk \cdot \mathrm{poly}(n), O(dkn^{1.8}\varepsilon(n^{0.1})))$-message indistinguishable.*

This claim follows by combining Claims 3.11, 3.15, and 3.18.

3.6 Known attacks to the scheme

To evaluate a circuit of depth d, our scheme requires keys of size $O((d \log d)n)$, where n is the security parameter. One important tool in our analysis is the length-preserving recryption circuit from Section 3.4. There we proved that recryption is secure provided it is used on independent key pairs. It is tempting to instantiate this construction over the same key pair, in the spirit of "circular security" prevalent in other works on homomorphic encryption. This would indeed eliminate the dependence on d (and also obviate the need for reducing the key error).

While we do not know if the suggested circular security assumption is valid or not, we are uncomfortable conjecturing it for the following reason. In the auxiliary key

information, every one of the n elements y_i of the "secret key vector" y is encoded by a ciphertext c_i of length n, so that all the ciphertexts decode without error. In view of the simplicity of our decryptions, we feel that if such a property holds at all, it should be achievable by direct construction (possibly using other reasonable security assumptions) rather than the somewhat complex mechanism of Section 3.3. We were not able to come up with such a direct construction without suffering a security flaw.

It turns out any encryption scheme with such a simple decryption function cannot be homomorphic at the same time. Brakerski [Bra13] shows if a scheme supports homomorphic evaluation of the majority function, then its decryption cannot be weakly-learnable (e.g. linear). As a consequence, he gives two specific attacks on the schemes **HOM** and **BASIC** to falsify conjecture 3.1.

On the other side, Gauthier, Otmani and Tillich [GOT12] observes that the public key in the scheme **K** can be viewed as a modified Reed-Solomon code obtained by planting a zero submatrix in the Vandermonde generating matrix defining it. The rows that define this submatrix are kept secret and form a set S. They next look at the "square code" generated by the pointwise products of codewords of the public key. By considering the dimension of the subcode obtained by projecting the square code onto a subset I of the rows, they are able to show that the dimension of the subcode is directly related to the cardinality of the intersection of I with S. This gives an attack which recovers the full set S, breaking **K** completely.

We now present the attacks on **BASIC** and **K** in detail. We say an attack *completely breaks* a scheme if there exists a randomized adversary that upon receiving the public key and $\mathbf{Enc}_{PK}(m)$ for arbitrary value of m, returns m with probability $1-o(1)$ in time polynomial in the security parameter.

3.6.1 An attack on BASIC using homomorphism

First we present Brakerski's attack on the scheme **BASIC**.

Theorem 3.21. *There exists a polynomial time attack that completely breaks* **BASIC**.

Proof. Consider an instantiation of $\mathbf{K}_q(n)$ with keys (PK, SK) and an instantiation of $\mathbf{K}_q(n')$ with keys (PK', SK'), for $n' = n^{1+\alpha}$. Let y (resp. y') be the designated solution to the system (3.1), which specifies the decryption space of $\mathbf{K}_q(n)$ (resp. $\mathbf{K}_q(n')$).

Let $H = H_{n':n} \in \mathbb{F}_q^{n' \times n}$ be an $n' \times n$ matrix that represents the auxiliary information $I(SK, PK')$, that is, the i-th column of $H_{n':n}$ is the encryption $\mathbf{Enc}_{PK'}(y_i)$. By Claim 3.12, with probability at least $1 - n^{-\Omega(1)}$, $H_{n':n}$ is good, in which case it holds

that

$$y'^T H = y^T \text{ and } Hc = \mathbf{ReEnc}_{I(SK,PK')}(c)$$

where $c \in Dec_{SK}(m)$ for every m. Note that the rank of H is at most n.

The adversary will be given H and the public key PK, and will be able to decrypt any vector $c = \mathbf{Enc}_{PK}(m)$ with high probability, namely compute $\langle y, c \rangle$.

The attack. As explained above, the input to the adversary is H, PK and challenge $c = \mathbf{Enc}_{PK}(m)$. The adversary will execute as follows:

1. Generate $k = n^{1+\varepsilon}$ encryptions of 0, denoted $v_1, \ldots, v_k$, for $\varepsilon = \alpha(1-\alpha)/4$.

2. For all $i = 1, \ldots, k$, compute $v'_i = Hv_i$ (the recryptions of the ciphertexts above through H). Also compute $o' = H\mathbf{1}$ (the recryption of the all-one vector).

3. Find a vector $\tilde{y'} \in \mathbb{F}_q^{n'}$ such that $\langle \tilde{y'}, v'_i \rangle = 0$ for all i, and $\langle \tilde{y'}, o' \rangle = 1$. Such a vector necessarily exists if all v_i's are in $Dec_{SK}(0)$, since y' is an example of such a vector.

4. Given a challenge ciphertext c, compute $c' = Hc$ and output $m = \langle \tilde{y'}, c' \rangle$ (namely, $m = \tilde{y'}Hc$).

Correctness. To analyze the correctness of the attack, we first notice that the space $Dec_{SK}(0)$ is linear (this is exactly the orthogonal space to y). Note that $Dec_{SK}(m) = Dec_{SK}(0) + m\mathbf{1}$. We recall that the space $Enc_{PK}(m) \subseteq Dec_{SK}(m)$ for every $m \in \Sigma$.

By the definition of recryption, the space $Enc_{PK'}(0)$ contains all vectors of the form Hz such that $z \in Dec_{SK}(0)$. This is a linear space with dimension at most n.

Consider the challenge ciphertext $c = \mathbf{Enc}_{PK}(m)$. We can think of c as an encryption of 0 with an added term $m\mathbf{1}$. We therefore denote $c = c_0 + m\mathbf{1}$. Again this yields a c'_0 such that $c' = c'_0 + mo'$.

Now consider the distribution D over $Enc_{PK}(0)$, which is the distribution of encryptions of 0 (i.e. the distribution $c = \mathbf{Enc}_{PK}(0)$, conditioned on $\langle y, c \rangle = 0$). The distribution D' is defined by projecting D through H. With probability $1 - n^{-\Omega(1)}$, it holds that $v'_1, \ldots, v'_k$ and c'_0 are uniform samples from D'.

It follows from Lemma 3.22 below that $c'_0 \in \text{span}\{v'_1, \ldots, v'_k\}$ with probability $1 - n^{-\Omega(1)}$. In such case

$$\langle \tilde{y'}, c' \rangle = \langle \tilde{y'}, c'_0 \rangle + m\langle \tilde{y'}, o' \rangle = m.$$

We conclude that with probability $1 - n^{-\Omega(1)}$, the adversary correctly decrypts c as required. □

Lemma 3.22. *Let D be a distribution over a linear space S of dimension s. For all k, define*

$$\delta_k = \Pr_{v_1,\ldots,v_k \sim S}[v_k \notin \text{span}\{v_1, \ldots, v_{k-1}\}].$$

Then $\delta_k \leq s/k$.

Proof. Notice that by symmetry $\delta_i \geq \delta_{i+1}$ for all i. Let D_i be a random variable that denotes the dimension of $\text{span}\{v_1, \ldots, v_i\}$. Note that always $D_i \leq s$.

Let I_i be the indicator variable denote the event $v_i \notin \text{span}\{v_1, \ldots, v_{i-1}\}$, note that $\delta_i = \Pr[I_i]$. By definition,

$$D_k = \sum_{i=1}^{k} I_i.$$

Therefore

$$s \geq \mathrm{E}[D_k] = \sum_{i=1}^{k} \Pr[I_i] = \sum_{i=1}^{k} \delta_i \geq k\delta_k,$$

and the lemma follows. □

3.6.2 A structural attack on K

We now present the attack by Gauthier et al. First notice the public key matrix P defined in (3.1) can be viewed as the generating matrix of the linear code

$$\mathcal{C} = \text{span}\{P_i \mid 1 \leq i \leq n\},$$

where P_i is the i-th column of P. Define the *square code* $\mathcal{C}^2$ of $\mathcal{C}$ to be

$$\mathcal{C}^2 = \text{span}\{P_i \odot P_j \mid 1 \leq i, j \leq n\},$$

where $\odot$ denotes the pointwise multplication over $\mathbb{F}_q^n$. Note that since $P = MR$ and R is invertible, we also have

$$\mathcal{C}^2 = \text{span}\{M_i \odot M_j \mid 1 \leq i, j \leq n\}.$$

We also define $\mathcal{C}_I^2$ to be the square code of the subcode $\mathcal{C}_I$ that is obtained by projecting $\mathcal{C}$ onto the coordinates restricted to I. The key observation of the attack is that when I does not intersect S, the dimension of $\mathcal{C}_I^2$ is exactly $2r$, while in the case I intersects S,

the dimension of C_I^2 increases by roughly $|I \cap S|$. More precisely, we have the following proposition.

Proposition 3.23. *Let $I \subseteq [n]$ and set $J = I \cap S$. Suppose $|J| \leq s/3 - 1$ and $|I| - |J| \geq 2r$. Then*

$$\dim(\mathcal{C}_I^2) = 2r - 1 + |J|.$$

Using this proposition, we can construct the following randomized distinguisher to recover S in polynomial time: Sample a random subset I of size $3r$, with probability $1 - n^{-\Omega(1)}$ the conditions on $|I|$ and $|J|$ in the proposition are satisfied. For each $i \in I$, consider $I' = I - \{i\}$. If $\dim(\mathcal{C}_{I'}^2) < \dim(\mathcal{C}_I^2)$, we conclude $i \in S$, otherwise $i \notin S$. To determine whether a position $i' \notin I$ belongs to S, we can exchange an element in I with i' and compare the dimensions of two subcodes.

The proposition follows from the next two lemmas, which give a basis for $\mathcal{C}_I^2$ that is of size $2r - 1 + |J|$.

For each $i \in I$, define $b_i = 0$ if $i \in J$ and $b_i = a_i$ otherwise. For each $t \in [r]$, let a^t and b^t be two vectors in $\mathbb{F}_q^I$ defined by $(a^t)_i = a_i^t$ and $(b^t)_i = b_i^t$ for every $i \in I$, respectively.

Lemma 3.24. *If t is an integer that satisfies $s/3 + |J| + 2 \leq t \leq 2s/3$, then $a^t \in \mathrm{span}\{a^{s/3+2}, \ldots, a^{t-1}, b^{s/3+2}, \ldots, b^t\}$.*

Proof. Let p be a polynomial over $\mathbb{F}_q$ defined by

$$p(x) = x^{t-|J|} \prod_{i \in J} (x - a_i) = \sum_{u=t-|J|}^{t} c_u x^u,$$

where each c_u is some element in $\mathbb{F}_q$. By assumption, p has degree at most $2s/3$.

For each $i \in J$, we have $p(a_i) = 0$ by construction and $p(b_i) = 0$ because $b_i = 0$. So $p(a_i) = p(b_i)$ for every $i \in J$. For each $i \notin J$, we also have $p(a_i) = p(b_i)$ because $a_i = b_i$. Therefore for every $i \in I$, we have

$$\sum_{u=t-|J|}^{t} c_u a_i^u = \sum_{u=t-|J|}^{t} c_u b_i^u.$$

Since $c_t = 1$, a^t can be written as

$$a^t = \sum_{u=t-|J|}^{t} c_u b^u - \sum_{u=t-|J|}^{t-1} c_u a^u,$$

□

Lemma 3.24 implies $\{a^2, \ldots, a^{s/3+|J|+1}, b^{s/3+2}, \ldots b^{2r}\}$ is a generating set of $\mathcal{C}_I^2$, the next lemma shows that it is linearly independent.

Lemma 3.25. *Suppose* $|J| \leq s/3 - 1$ *and* $|I| - |J| \geq 2r$. *Then*

$$B = \{a^2, \ldots, a^{s/3+|J|+1}, b^{s/3+2}, \ldots b^{2r}\}$$

is a basis for $\mathcal{C}_I^2$.

Proof. It remains to show that B is linear independent. Suppose

$$\sum_{t=2}^{s/3+|J|+1} \alpha_t a^t + \sum_{t=s/3+2}^{2r} \beta_t b^t = 0$$

By setting $\alpha_t = 0$ for $s/3 + |J| + 2 \leq t \leq 2r$ and $\beta_t = 0$ for $2 \leq t \leq s/3 + 1$, we have

$$\sum_{t=2}^{2r} (\alpha_t a^t + \beta_t b^t) = 0.$$

Let $p(x) = \sum_{t=2}^{2r} (\alpha_t + \beta_t) x^t$ be a degree $2r$ polynomial. Since $a_i^t = b_i^t$ for $i \notin J$ and $|I| - |J| \geq 2r$, we have $p \equiv 0$ and so $\alpha_t = -\beta_t$ for every t. Therefore, $\alpha_i^t = \beta_i^t = 0$ for $2 \leq t \leq s/3 + 1$ and $s/3 + |J| + 2 \leq t \leq 2r$.

In the case of $s/3 + 2 \leq t \leq s/3 + |J| + 1$, we have $b_i^t = 0$ for $i \in J$. Hence for $i \in J$ we have

$$\sum_{t=s/3+2}^{s/3+|J|+1} \alpha_t a_i^t = 0.$$

Let $q(x) = \sum_{t=s/3+2}^{s/3+|J|+1} \alpha_t x^t = x^{s/3+2} r(x)$, where $r(x) = \sum_{t=s/3+2}^{s/3+|J|+1} \alpha_t x^{t-s/3-2}$ is a degree $|J| - 1$ polynomial. Since $q(a_i) = 0$ and $a_i \neq 0$ for every $i \in J$, we have $r \equiv 0$ and so $a_i^t = 0$ for $s/3 + 2 \leq t \leq s/3 + |J| + 1$. Therefore we have $\alpha_t = \beta_t = 0$ for every t. □

Chapter 4

On the depth complexity of homomorphic encryption schemes

In this chapter we give concrete complexity-theoretic evidence that encryption schemes that support homomorphic evaluation of essentially any non-trivial functionality are more complex than ordinary encryption schemes. Our main result (Theorem 4.2) shows that homomorphic evaluation of any non-trivial functionality (for example the AND function) that depends on sufficiently many inputs cannot be implemented by circuits of constant depth and subexponential size with respect to any CPA secure encryption scheme. In Section 4.3 we review some proposals of CPA secure private key encryption schemes of quasipolynomial security that can be implemented in this model. In the public key setting, we observe that the cryptosystem of Applebaum, Barak, and Wigderson [ABW10] can be implemented in constant depth.

Thus constant-depth circuits provide sufficient computational power for implementing both private and public-key encryption schemes (under previously studied assumptions), but not variants of such schemes that support homomorphic evaluation of any non-trivial functionality.

4.1 Definitions

Let us recall the definitions of homomorphic evaluation in Chapter 2. For simplicity, we assume the ciphertext set is over $\{0,1\}^n$ and state everything in the non-uniform setting.

Definition 4.1. Let $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ be a private-key encryption scheme over message set Σ with ciphertexts in $\{0,1\}^n$. We say a circuit H is a *homomorphic evaluator* of

$f : \Sigma^k \to \Sigma$ with error δ if (1) the output length of H is bounded by a function that depends only on the security parameter and (2) for all $m_1, \ldots, m_k \in \Sigma$,

$$\Pr[\mathbf{Dec}_{SK}(H(\mathbf{Enc}_{SK}(m_1, R_1), \ldots, \mathbf{Enc}_{SK}(m_k, R_k))) = f(m_1, \ldots, m_k)] \geq 1 - \delta,$$

where $SK \sim \mathbf{Gen}$ is a uniformly chosen secret key and $R_1, \ldots, R_k$ are independent random seeds.

In the public-key setting, we are given an encryption scheme $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ and require that

$$\Pr[\mathbf{Dec}_{SK}(H(PK, \mathbf{Enc}_{PK}(m_1, R_1), \ldots, \mathbf{Enc}_{PK}(m_k, R_k))) = f(m_1, \ldots, m_k)] \geq 1 - \delta.$$

where $(PK, SK) \sim \mathbf{Gen}$ is a random key pair.

Our negative result will only apply to functions whose number of relevant inputs k is sufficiently large in terms of n. Beyond this requirement, we do not make any assumption on f.

The requirement we make on the encryption scheme is CPA message indistinguishability. A private-key encryption scheme is (s, d, ε) CPA message indistinguishable if for every pair of messages $m, m' \in \Sigma$ and every distinguishing oracle circuit $D^?$ of size s and depth d,

$$|\Pr_{SK,R}[D^{\mathbf{Enc}(SK,\cdot)}(\mathbf{Enc}_{SK}(m, R)) = 1] - \Pr_{SK,R}[D^{\mathbf{Enc}(SK,\cdot)}(\mathbf{Enc}_{SK}(m', R)) = 1]| \leq \varepsilon.$$

In the public key setting CPA security follows from ordinary message indistinguishability:

$$|\Pr_{PK,R}[D(PK, \mathbf{Enc}_{PK}(m, R)) = 1] - \Pr_{PK,R}[D(PK, \mathbf{Enc}_{PK}(m', R)) = 1]| \leq \varepsilon.$$

4.2 Homomorphic evaluation requires depth

Theorem 4.2. *Suppose* $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ *is an* $(2s + k + O(1), d + 1, 1/6(k + 1))$ *CPA message indistinguishable private-key (resp. public-key) encryption scheme. Let* H *be a homomorphic evaluator of size* s *and depth* d *with error at most* $1/3$ *for some* $f : \Sigma^k \to \Sigma$ *that depends on all of its inputs with respect to this scheme. Then* $s > 2^{\Omega((k/6n)^{1/(d-1)})}$.

For notational simplicity, we present the proof for the private key variant. Since f depends on all its inputs, for every $i \in [k]$ there is a pair of messages m and m' that

differ only in coordinate i such that $f(m) \neq f(m')$. Now suppose H is a homomorphic evaluator for f with error $1/3$. Then

$$\Pr[\mathbf{Dec}(H(\mathbf{Enc}(m_1, R_1), \ldots, \mathbf{Enc}(m_i, R_i), \ldots, \mathbf{Enc}(m_k, R_k))) \neq f(m)] \leq 1/3 \quad \text{and}$$
$$\Pr[\mathbf{Dec}(H(\mathbf{Enc}(m_1, R_1), \ldots, \mathbf{Enc}(m'_i, R'_i), \ldots, \mathbf{Enc}(m_k, R_k))) \neq f(m')] \leq 1/3,$$

where the probability is taken over the choice of secret key SK (which we omit to simplify notation) and the randomness $R_1, \ldots, R_i, R'_i, \ldots, R_k$ used in the encryption. Since $f(m) \neq f(m')$, it follows that

$$\Pr[\mathbf{Dec}(H(\mathbf{Enc}(m_1, R_1), \ldots, \mathbf{Enc}(m_i, R_i), \ldots, \mathbf{Enc}(m_k, R_k))) \neq \mathbf{Dec}(H(\mathbf{Enc}(m_1, R_1), \ldots, \mathbf{Enc}(m'_i, R'_i), \ldots, \mathbf{Enc}(m_k, R_k)))] \geq 1/3.$$

Therefore it must be that

$$\Pr[H(\mathbf{Enc}(m_1, R_1), \ldots, \mathbf{Enc}(m_i, R_i), \ldots, \mathbf{Enc}(m_k, R_k)) \neq H(\mathbf{Enc}(m_1, R_1), \ldots, \mathbf{Enc}(m'_i, R'_i), \ldots, \mathbf{Enc}(m_k, R_k))] \geq 1/3.$$

By CPA message indistinguishability and a hybrid argument, we can replace $m_1, \ldots, m_i, m'_i, \ldots, m_k$ by 0 to obtain

$$\Pr[H(\mathbf{Enc}(0, R_1), \ldots, \mathbf{Enc}(0, R_i), \ldots, \mathbf{Enc}(0, R_k)) \neq H(\mathbf{Enc}(0, R_1), \ldots, \mathbf{Enc}(0, R'_i), \ldots, \mathbf{Enc}(0, R_k))] \geq 1/6. \quad (4.1)$$

Lemma 4.3. *Let $D_1, \ldots, D_k$ be any distributions over $\{0,1\}^n$. Let $g : (\{0,1\}^n)^k \to \{0,1\}$ be a circuit of size s and depth d where $s \leq 2^{(\varepsilon k)^{1/(d-1)}/K}$ for some absolute constant K. Then*

$$\Pr[g(X_1, \ldots, X_i, \ldots, X_k) \neq g(X_1, \ldots, X'_i, \ldots, X_k)] < \varepsilon$$

where the randomness is taken over the choice of $i \sim [k]$ and independent samples $X_1 \sim D_1, \ldots, X_i, X'_i \sim D_i, \ldots, X_k \sim D_k$.

We apply this Lemma with D_i equal to the distribution of encryptions of 0 and $\varepsilon = 1/6n$ to each of the n outputs of H and take a union bound to conclude that (4.1) is violated unless $s > 2^{\Omega((k/6n)^{1/(d-1)})}$.

Proof of Lemma 4.3. Fix any pair $Z, Z' \in (\{0,1\}^n)^k$. For any $w \in \{0,1\}^k$, let $Z_w \in$

$(\{0,1\}^n)^k$ be the string such that

$$\text{the } i\text{-th block of } Z_w = \begin{cases} \text{the } i\text{-th block of } Z, & \text{if } w_i = 0 \\ \text{the } i\text{-th block of } Z', & \text{if } w_i = 1. \end{cases}$$

Let $h_{Z,Z'}(w) = g(Z_w)$. Then h is of size at most s and depth at most d. By Boppana [Bop97], for every Z and Z' we have

$$\Pr_{W,i}[h_{Z,Z'}(W) \neq h_{Z,Z'}(W + e_i)] \leq (K \log s)^{d-1}/k$$

for some constant K. Therefore for Z, Z' sampled independently from $D_1 \times \cdots \times D_k$ we have

$$\begin{aligned} \Pr[g(X_1, \ldots, X_i, \ldots, X_k) \neq g(X_1, \ldots, X_i', \ldots, X_k)] &= \mathrm{E}_{Z,Z'}[\Pr_{W,i}[h_{Z,Z'}(W) \neq h_{Z,Z'}(W + e_i)]] \\ &= \mathrm{E}_{Z,Z'}[(K \log s)^{d-1}/k] \\ &= (K \log s)^{d-1}/k. \end{aligned}$$

It follows that if this probability is at most ε, then $s \leq 2^{(\varepsilon k)^{1/(d-1)}/K}$. □

A similar lemma was proved by Blais, O'Donnell, and Wimmer [BOW10] for noise sensitivity of boolean functions. Here we adapted their argument to influence.

4.3 On CPA secure encryption schemes in AC^0

In this section we review the depth complexity of some studied candidate CPA secure encryption schemes. To begin with, we observe that asymptotically superpolynomial security cannot be achieved by NC^0 decryption circuits: If every output of the decryption circuit depends on at most d bits of the ciphertext, then for any message m the decryption circuit on the distribution of encryptions of m can be PAC-learned in time $O_d(n^d)$, violating CPA security.

Kharitonov [Kha93] implicitly shows the existence of a "weakly pseudorandom" function family in AC^0 that is $2^{\mathrm{poly}\log n}$ hard to predict on a uniformly random input even from membership queries (assuming Blum integers are sufficiently hard to factor). This function family can be used to obtain a CPA secure symmetric key encryption scheme whose encryption and decryption algorithms are in AC^0. However, we do not know if key generation (which involves generating random Blum integers of magnitude $2^{\mathrm{poly}\log n}$) can be performed in AC^0. Gilbert et al. [GRS08] give a probabilistic CPA secure symmetric

key encryption scheme whose security can be reduced to the hardness of the Learning Parity with Noise (LPN) problem. The current best known algorithms for the LPN problem over $\{0,1\}^m$ all run in time $2^{\Theta(m/\log m)}$. Assuming this is optimal, by setting $m = (\log n)^d$ one can implement all components of this scheme using circuits of size $\mathrm{poly}(n)$ and depth $d + O(1)$, and the scheme has security $2^{\Theta((\log n)^d/\log\log n)}$.

We are not aware of any implementation of a public key encryption scheme with all but negligible security all of whose components are in AC^0. Here we show that the cryptosystem proposed by Applebaum, Barak and Wigderson [ABW10] can be implemented using circuits of polynomial size and constant depth in the security parameter. The variant of the cryptosystem we discuss is conjectured to have security $n^{\Omega(\log n)}$.[1]

First we review the key generation, encryption and decryption in the ABW encryption scheme. One can refer to [ABW10] for further details. Then we show how to implement each operation in constant depth.

The public key is a random bipartite graph $G = ((U,V),E)$, where $|U| = n$ and $|V| = r = n^{0.9}$, generated in the following way. First choose a random subset $S \subseteq U$ and $T \subseteq V$ of size s and $s/3$ respectively, where $s = O(\log n)$. Each vertex in S is connected to d (possibly repeated) random vertices in T and each vertex outside S is connected to d random vertices in V. The secret key SK is an odd size subset of S such that each vertex in T has an even number of neighbors in SK.

To encrypt a message $m \in \{0,1\}$, choose a random subset T' of V and output $y + e + m\mathbf{1}$, where each coordinate of $y \in \{0,1\}^n$ is the degree of the corresponding vertex in U restricted to $T' \bmod 2$, $e \in \{0,1\}^n$ is a vector with each coordinate sampled from a distribution $\hat{\eta}$ with $\Pr[\hat{\eta} = 0] = \eta$ independently, and $\mathbf{1} \in \{0,1\}^n$ is the all ones vector.

To decrypt a ciphertext $c \in \{0,1\}^n$, output $\sum_{i \in SK} c_i$. Now we give an AC^0 implementation of the cryptosystem.

Implementation of the ABW cryptosystem in AC^0

Key Generation: Sample

1. $y_1, y_2, \ldots, y_s$ from $[n]$ and $w_1, w_2, \ldots, w_{s/3}$ from $[r]$ to represent the subsets $S \subseteq U$ and $T \subseteq V$, respectively;

2. $v_{i,1}, \ldots, v_{i,d}$ from $[r]$ for every i from 1 to n. These are the random neighbors of each vertex i in $U \setminus S$;

[1] Owing to the existence of a quasipolynomial time algorithm for learning from random examples [LMN93], if ciphertexts are computationally indistinguishable from the uniform distribution, any AC^0 decryption algorithm can be broken in time $2^{\mathrm{poly}\log n}$.

3. $\hat{v}_{i,1}, \ldots, \hat{v}_{i,d}$ from $[s/3]$ for every i from 1 to s. These become the random neighbors of the vertices in S after being mapped to the w_i's by the index function $\iota : [s/3] \to [r]$ such that $\iota(i) = w_i$. This function can be written as

$$\iota(i) = \bigvee_{j=1}^{s/3} [(i = j) \wedge w_j].$$

The key generation circuit outputs $v_{i,1}, \ldots v_{i,d}$ if the vertex i is not in S, and outputs $\iota(\hat{v}_{i,1}), \iota(\hat{v}_{i,2}), \ldots, \iota(\hat{v}_{i,d})$ otherwise. Now we can output the jth random neighbor of each vertex $i \in U$ by

$$\left[\delta_i \wedge \bigvee_{k=1}^{s} [(i = y_k) \wedge \iota(\hat{v}_{k,j})]\right] \vee (\overline{\delta_i} \wedge v_{i,j}),$$

where $\delta_i := \bigvee_{k=1}^{s} (i = y_k)$ indicates whether i belongs to S.

To come up with the secret key SK, we enumerate all the possible subsets of S (recall that $s = O(\log n)$) and output the first one that satisfies the linear dependency. Given an odd size subset of S indicated by the support of the vector $a \in \{0,1\}^s$. It is not difficult to see that the formula

$$f_a = \bigvee_{j=1}^{s/3} \bigoplus_{i:a_i=1} \bigoplus_{k=1}^{d} (\hat{v}_{i,k} = j)$$

outputs 0 if every vertex in T has an even number of neighbors in the support of a and outputs 1 otherwise. (Since the XOR involves only $O(d \log n)$ inputs, it can be implemented in depth two and size $n^{O(d)}$.) Thus we can enumerate all the possible $a \in \{0,1\}^s$ of odd hamming weight and output the first subset a with $f_a = 0$. The secret key is represented by a vector z containing s entries in $[n]$, where each nonzero entry corresponds to a vertex in SK. More precisely, we output the ith entry as

$$z_i = \iota\left(\bigvee_{a \in \{0,1\}^s : wt(a) \text{ is odd}} \left[\overline{f_a} \wedge \left(\bigwedge_{b<a} f_b\right) \wedge (a_i \wedge i)\right]\right).$$

Encryption: Given a public key represented by the neighbors $v_{i,1}, \ldots, v_{i,d}$ of each vertex i in U. To encrypt a message $m \in \{0,1\}$, choose a random vector x in $\{0,1\}^r$ whose support forms the subset T' of V, a noise vector $e \in \{0,1\}^n$ by choosing each of its entries independently from $\hat{\eta}$. The ith bit of the encryption can be written as

$$c_i = \bigvee_{\substack{k_i \neq k_j, 1 \leq i < j \leq d, k_i \in [r] \\ a_1, \ldots, a_d:\ a_1 + \cdots + a_d = 1}} [\bigwedge_{j=1}^{d} (v_{i,j} = k_j) \wedge (x_{k_1} = a_1) \wedge \cdots \wedge (x_{k_d} = a_d)] \oplus e_i \oplus m.$$

Decryption: Given a ciphertext c and the secret key SK represented by the vector $z \in \{0,1\}^{s \times \log n}$, output

$$\bigoplus_{i=1}^{s} \bigvee_{k=1}^{n} [(z_i = k) \wedge c_k].$$

Reducing the encryption error The ABW cryptosystem (as well as the LPN-based system of Gilbert et al.) has noticeable encryption error. The encryption error can be made negligible by encrypting the message independently multiple times. While some of the multiple encryptions may be erroneous, with all but negligible probability at least 2/3 of them will be correct. The errors can be corrected by taking approximate majority at the decryption stage, which can be implemented using circuits of depth 3 [Ajt96], thereby preserving the constant depth complexity of the implementation.

Chapter 5

Limits of provable security for homomorphic encryption

A promise problem $\Pi = (\Pi_Y, \Pi_N)$ has an *interactive proof* if there is a randomized polynomial-time verifier V that on input x, can exchange at most $\text{poly}(|x|)$ messages with any computational unbounded prover P such that it satisfies the following conditions:

1. If $x \in \Pi_Y$, there exists an honest prover P that makes V accept with probability at least $2/3$.

2. If $x \in \Pi_N$, for any prover P, V accepts with probability at most $1/3$.

We say Π is in AM, if the verifier reveals the randomness used in its computation and exchanges only constant number of messages with the prover .

In this chapter, our main theorem (Theorem 5.4) shows that any public key encryption scheme that supports efficient weak homomorphic evaluation of any sufficiently "sensitive" collection of functions cannot be proved message indistinguishable beyond $\text{AM} \cap \text{coAM}$, even under adaptive reductions. Examples of such functions are parities, majorities, and the collection of all AND and OR functions.

Examples of encryption schemes that our result applies to include El Gamal encryption [Gam85], Paillier encryption [Pai99], as well as the more recent somewhat and fully homomorphic encryption schemes of Gentry [Gen09b], Van Dijk et al. [vDGHV10], and Brakerski and Vaikuntanathan [BV11] (which build upon the lattice-based cryptosystems of Regev [Reg09] and Peikert [Pei09]).

In Theorem 5.5 we show that if the reduction has constant query complexity, then message indistinguishability cannot be proved beyond statistical zero knowledge (SZK)[1],

[1]For a formal definition of SZK, we refer the readers to [Gol08].

which is a subclass of AM $\cap$ coAM.

The reductions we consider are randomized and meet the following definition: Given an input, the reduction makes arbitrary (adaptive) queries to a distinguishing oracle for bit encryptions. We require that for any (not necessarily efficient) distinguishing oracle, which may depend on the input to the reduction, the reduction outputs the correct answer. We do not know of any cryptographic reductions that treat the adversary as a black box which fall outside our definition.

Lemma 5.8, which is used in the proofs of Theorems 5.4 and 5.5, gives a way to obtain rerandomization of ciphertexts from any homomorphic evaluator for the function of interest. While rerandomization has been used in constructions of homomorphic evaluators [Gen09b, vDGHV10], it is not a priori clear that it is necessary for homomorphic evaluation. Homomorphic evaluation may be implemented deterministically while rerandomization requires randomness.

The statistical error of the rerandomization in Lemma 5.8 is noticeable. While this is sufficient for our main application, a negligible error would be desirable for most applications of rerandomization in cryptography. In Theorem 5.6 we show a transformation of a strong homomorphic evaluator for almost any function into a rerandomization that preserves negligible statistical error. Essentially the only exceptions to which our result does not apply are that AND, OR, and NOT functions.

5.1 Overview of the proof

From homomorphic evaluation to rerandomization (Section 5.4) To begin with let's assume that we have a *strong* (i.e., distribution-preserving) homomorphic evaluator H for the majority function maj_n on n inputs. This is an algorithm that takes as inputs independent encryptions of $x_1, \ldots, x_n$ and outputs a ciphertext which is statistically close to an encryption of $maj_n(x_1, \ldots, x_n)$. We show that H can be used to obtain an approximate *rerandomization* **Rer**: This is a procedure that takes an encryption as its input and produces an independent and identically distributed encryption as its output. Our rerandomization will be approximate in the sense that the input and output of **Rer** will be only statistically close to independent.

One way to obtain rerandomization is as follows: Given a ciphertext C, generate $(n-1)/2$ independent encryptions of 0, $(n-1)/2$ independent encryptions of 1, randomly shuffle them together with C and feed the n resulting ciphertexts to the homomorphic evaluator for majority. By the strong homomorphic property, the output of the homomorphic evaluator will be identically distributed with C. But why should they be

independent? From the point of view of the homomorphic evaluator, if C is an encryption of b, then it is indistinguishable from the other $(n-1)/2$ encryptions of b. Since the output of the homomorphic evaluator is bounded in length, the evaluator must "forget" most of the information about most of the ciphertexts it is given as inputs, including C as it is indistinguishable from the others. Therefore the output is forced to look almost statistically independent of C.

In Lemma 5.8 we generalize this argument to a much wider class of functions which we call *sensitive* (see Section 5.2) and to weak (i.e., compact) homomorphic evaluators, in which case we obtain a weaker notion of rerandomization.

A strong rerandomization procedure can be used to distinguish encryptions in statistical zero-knowledge by reduction to the "statistical distance" problem: A rerandomized encryption of 0 is statistically close to an encryption of 0, but statistically far from an encryption of 1. Mahmoody and Xiao's simulation of $\mathrm{BPP}^{\mathrm{SZK}}$ in AM [MX10] can then be used to emulate the reduction by a proof system. When only weak one-sided rerandomization is available, it is not clear that encryptions are distinguishable in statistical zero-knowledge, and we construct a somewhat different proof system. For the sake of clarity, however, in the rest of this discussion we will assume the availability of strong rerandomization.

From rerandomization to a distinguishing protocol (Section 5.5) To turn a reduction from distinguishing encryptions to L into a proof system for $\overline{L}$, we proceed as in previous works: The verifier plays the role of the reduction and the prover plays the role of the distinguishing oracle. The challenge is to force the prover to give answers that are consistent with a specific, fixed distinguishing oracle.

To illustrate the difficulties in the context of public key encryption, let us point out the deficiencies of some naive proof systems. Suppose the verifier submits a public key-ciphertext query (PK, C) to the prover, who is supposed to act as a distinguishing oracle. A natural attempt is to ask the prover to provide the message m and randomness R such that C is an encryption of m under public key PK with randomness R. This fails to account for the possibility that C may not be a valid ciphertext at all: Perhaps there is no pair (m, R) that encrypts to C under PK. It is not clear how a prover can certify such a statement. Another attempt would be to ask the prover for the secret key SK associated to PK. Again, it is not clear how to achieve completeness in case the public key is invalid and there is no corresponding secret key, or soundness in case the public key can be paired with several different secret keys (the choice of which may affect how different invalid ciphertexts decrypt).

Our protocol works as follows: Given a query (PK, C), the verifier asks the prover for the value b that encrypts to C, together with a proof that the rerandomization of C is statistically close to encryptions of b but statistically far from encryptions of $\overline{b}$. If the pair (PK, C) is properly distributed, this forces the prover to give a unique correct answer. But since statistical closeness and statistical farness are both efficiently verifiable [BBM11, SV03], the prover can now also certify that a pair (PK, C) is *not* a valid public key-ciphertext pair. We call this protocol DP (the distinguishing protocol).

One important detail is that the protocols for statistical closeness and statistical farness are only guaranteed to solve promise versions of these problems: For a given gap $[\ell, r)$, they can distinguish distributions that are within statistical distance ℓ from those that are at distance at least r, but give no guarantee about the outcome for instances that fall inside the gap. Therefore DP is only complete and sound provided that none of the underlying instances fall inside the respective gaps.

The proof system (Section 5.7) Given a reduction R from a decision problem L to distinguishing encryptions, this suggests the following constant-round proof system for $\overline{L}$: On a given input, the verifier chooses randomness for the reduction and sends this randomness to the prover. The prover sends back a transcript of the reduction interacting with a distinguishing oracle, which includes a list of queries (PK_i, C_i) made by the reduction together with an answer a_i saying if C_i encrypts 0 or 1 under PK_i, or the pair (PK_i, C_i) is invalid ($\perp$). The verifier and prover then apply the DP protocol to certify that all the answers a_i are correct.

This proof system is complete and sound, provided that all the inputs (PK_i, C_i, a_i) to the DP protocol satisfy its promise. But in general the verifier does not know in advance if the promise is satisfied or not. We resolve this issue by choosing the width of the gaps $[\ell, r)$ to be sufficiently small and by having the verifier randomize the location of the gaps. This should make it unlikely for any of the queries to fall inside the promise gap of DP.

This approach was also used by Bogdanov and Trevisan [BT06] in the context of non-adaptive reductions. An additional twist is required when the reduction is adaptive because the location of the gaps may affect the answers of the honest prover. For example, imagine an adaptive reduction that does a "binary search" for the gap $[\ell, r)$: If the first answer a is to the right of r, its next query will be $a/2$, and so on until it hits the gap. To handle such reductions, we want to make the location of the gaps in each round independent of the answers of the honest prover in the previous rounds. On the other hand, the locations of these gaps must be consistent with a specific, fixed

distinguishing oracle that the prover is required to emulate.

To achieve both objectives we specify a randomized family of distinguishing oracles, where for each query to the oracle the gap location is random, and the gap locations among the various queries are q-wise independent, where q is an upper bound on the number of queries performed by the reduction. In the first round of the reduction the verifier chooses a random oracle from this family and sends its (polynomial length) description to the prover. The honest prover is then expected to give answers that are consistent with this instantiation of the distinguishing oracle. By independence, the probability that any of the queries made by the honest prover falls inside the gap will be small. In Section 5.6.1 we develop the relevant complexity-theoretic framework and we prove Theorem 5.4 in Section 5.7.1.

To prevent any of the queries from falling into the gaps $[\ell, r)$, the size of the gaps needs to be inverse proportional to the number of queries made by the reduction. Unless the reduction makes a bounded number of queries, this requires protocols for statistical closeness and statistical farness where the verifier runs in time inverse polynomial to the size of the gap and the gap can be at an arbitrary location. Such protocols were developed by Bhatnagar, Bogdanov, and Mossel [BBM11][2] and we use them in the proof of Theorem 5.4.[3]

For reductions that make a constant number of queries, it is sufficient to have statistical closeness/farness protocols over a constant number of disjoint gaps $[\ell, r)$. Sahai and Vadhan [SV03] give implementations of such protocols in SZK. Using their protocols and the closure properties of SZK which we recall in Section 5.6.2, we prove Theorem 5.5 in Section 5.7.2.

Better rerandomization from strong homomorphic evaluation The rerandomization procedure we described above comes with a non-negligible statistical error. It is not difficult to construct examples showing that this error is inherent, even if the homomorphic evaluation is perfect, i.e. it induces no statistical error. In Section 5.8 we show that the statistical error can be reduced exponentially by iteratively applying the rerandomization on its output, provided f is not "exceptional". This proves Theorem 5.6.

[2]Technically their statement is not as strong as the one we need here, but their proof can be easily adapted. We provide the details in Appendix C.

[3]Similar issues arise in the work of Mahmoody and Xiao [MX10]. They work with the SZK-complete problem entropy difference. While their proof can be adapted to our setting, we find it more natural to work directly with instances of statistical difference.

5.2 Definitions

In this section we give definitions of homomorphic evaluation and rerandomization. We will use two notions of homomorphic evaluation. Recall in Chapter 2 we give a fairly weak definition of homomomorphic evaluation. This corresponds to weak (or compact) homomorphic evaluation in this chapter. In this section we give the corresponding strong notions. For the sake of clarify, we give the definition together with the weak one.

Homomorphic evaluation and rerandomization Let $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ be a bit encryption scheme. Fix a security parameter s and let $(PK, SK) \sim \mathbf{Gen}(1^s)$ the distribution on key pairs. (We will assume that s is implicit in the public and secret keys.)

Definition 5.1. Let $f\colon \{0,1\}^n \to \{0,1\}$ be a boolean function. We say H is a *strong homomorphic evaluator* for f with error ε if for all m in the domain of f, the random variables

$$(PK, H_{PK}(\mathbf{Enc}_{PK}(m_1), \ldots, \mathbf{Enc}_{PK}(m_n))) \quad \text{and} \quad (PK, \mathbf{Enc}_{PK}(f(m)))$$

(where all encryptions are independent) are within statistical distance ε.

This definition extends to functions from $\{0,1\}^* \to \{0,1\}$ in a straightforward way. We omit the details.

Definition 5.2. Let $f\colon \{0,1\}^* \to \{0,1\}$ be a boolean function. We say H is a *weak homomorphic evaluator* for f with error ε if (1) the output length of H is bounded by a function that depends only on the security parameter and (2) for all n and $m \in \{0,1\}^n$ in the domain of f,

$$\Pr[\mathbf{Dec}_{SK}(PK, H_{PK}(\mathbf{Enc}_{PK}(m_1), \ldots, \mathbf{Enc}_{PK}(m_n))) = f(m)] \geq 1 - \varepsilon,$$

where all encryptions are independent.[4]

A bit encryption scheme is *efficient* if $\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec}$ all run in time polynomial in the security parameter s. A homomorphic evaluator H is efficient if it is computable in time polynomial in s and n and its output length is polynomially bounded in s.

[4]Some works adopt the terms "distribution preserving" and "compact" homomorphic evaluation. We prefer the terms "strong" and "weak" for this chapter, as we are concerned with questions of computational complexity.

Definition 5.3. Let **Rer** be a randomized function that takes as input a public key and a ciphertext. In the following definitions R and R' are independent choices of randomness for **Rer**.

- We say **Rer** is a *strong rerandomization* with error ε if for every $m \in \{0,1\}$, the random variables

$$(PK, E, \mathbf{Rer}_{PK}(E,R)) \quad \text{and} \quad (PK, E, E')$$

  where $E, E' \sim \mathbf{Enc}_{PK}(m)$ are independent are within statistical distance ε.

- For $b \in \{0,1\}$, we say $\mathbf{Rer}^b$ is a *one-sided weak rerandomization* with decryption error ε and rerandomization error ρ if for every $m \in \{0,1\}$,

$$\Pr[\mathbf{Dec}_{SK}(\mathbf{Rer}^b_{PK}(\mathbf{Enc}_{PK}(m))) = m] \geq 1 - \varepsilon$$

  and the random variables

$$(PK, \mathbf{Rer}^b_{PK}(E,R), \mathbf{Rer}^b_{PK}(E,R')) \quad \text{and} \quad (PK, \mathbf{Rer}^b_{PK}(E,R), \mathbf{Rer}^b_{PK}(E',R'))$$

  where $E, E' \sim \mathbf{Enc}_{PK}(b)$ are independent are within statistical distance ρ.

We say the rerandomization is *efficient* if it can be evaluated in time polynomial in the security parameter.

Sensitivity of boolean functions We will use the following notion of sensitivity for boolean functions. For $x \in \{0,1\}^k$ let $x|_i$ be the string obtained by flipping the i-th bit of x and leaving the others unchanged. Let $f\colon \{0,1\}^k \to \{0,1\}$ be a boolean function and $b \in \{0,1\}$. We say f has *b-sensitivity* at least s if there exists an input $x \in \{0,1\}^k$ and a set $S \subseteq [k]$ of size s such that $f(x) = b$, $x_i = b$ for every $i \in S$, and $f(x|_i) = \bar{b}$ for every $i \in S$. We call (x, S) a witness that f has b-sensitivity at least s.

We say a family of functions $f = \{f_k\colon \{0,1\}^k \to \{0,1\}\}$ has *certifiable polynomial b-sensitivity* if there exists a constant $\alpha > 0$ so that on input k we can compute in time polynomial in k a witness that f_k has b-sensitivity at least k^α.

Examples of functions that have certifiable polynomial 0-sensitivity and 1-sensitivity include parity and majority. The AND function has certifiable polynomial 0-sensitivity while the OR function has certifiable polynomial 1-sensitivity.

Examples of functions whose 0-sensitivity and 1-sensitivity is less than s are functions that depend on at most $s-1$ of their inputs, i.e. $(s-1)$-juntas. Simon [Sim82] gives an

example of a function on k bits that depends on all its inputs but has 0-sensitivity and 1-sensitivity $O(\log k)$.

5.3 The main theorems

We say $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ supports weak homomorphic evaluation of f with error ε if it has an efficient homomorphic evaluator for f with error ε.

A *γ-distinguishing oracle* for $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ is a function D such that

$$\Pr[D(PK, \mathbf{Enc}_{PK}(0)) \text{ accepts}] - \Pr[D(PK, \mathbf{Enc}_{PK}(1)) \text{ accepts}] > \gamma.$$

A *reduction* from a decision problem L to γ-distinguishing encryptions in $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ is an efficient randomized oracle algorithm $R^?$ such that for every valid input x there exists a γ-distinguishing oracle D such that $R^D(x) = L(x)$ with probability at least $8/9$. (For our results the exact constant won't matter, as long as it is strictly greater than $1/2$.)

Theorem 5.4. *Let f_0 and f_1 be functions with certifiable polynomial 0-sensitivity and 1-sensitivity respectively (possibly the same function). Let $\varepsilon \in (0, 1/18)$ be any constant and $\delta \geq 2\sqrt{\varepsilon}$. Let $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ be a public key encryption scheme that supports efficient homomorphic evaluations of both f_0 and f_1 with error at most ε. If there is a reduction from L to $(1-\delta)$-distinguishing $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$, then L is in $\mathrm{AM} \cap \mathrm{coAM}$.*

We will assume that the reduction is *query length regular*: On input x, the reduction first computes a query length $m \geq |x|$ and only makes queries of length m. The theorem can be proved without this assumption, but we make it for notational convenience.

In the case when the reduction has constant query complexity, a stronger conclusion can be obtained.

Theorem 5.5. *Let f_0 and f_1 be functions with certifiable polynomial 0-sensitivity and 1-sensitivity respectively (possibly the same function). Let q be any constant, $\delta > 0$, and $\varepsilon = \varepsilon(q, \delta)$ a sufficiently small constant. Let $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ be a public key encryption scheme that supports efficient homomorphic evaluations of f_0 and f_1 with error at most ε. If there is a reduction from L to $(1-\delta)$-distinguishing $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ that makes at most q queries, then L is in statistical zero-knowledge.*

In particular, Theorems 5.4 and 5.5 apply to the following cases: (1) f_0 and f_1 are the parity function; (2) f_0 and f_1 are the majority function; (3) f_0 is OR and f_1 is AND.

Ron Rothblum [Rot11] shows how to turn a private-key encryption scheme into a public-key one using a homomorphic evaluator for parity. Combining the two results, the conclusions of Theorems 5.4 and 5.5 can be extended to private-key encryption schemes that support homomorphic evaluation of parity.

Our last result shows how to obtain strong rerandomization given a homomorphic evaluator for almost any function. We call a function $f\colon \{0,1\}^n \to \{0,1\}$ *exceptional* if it is one of the following functions of the inputs that it depends on: the constant 0, the constant 1, the identity, the NOT function, the AND function, the OR function.

Theorem 5.6. *Assume $f\colon \{0,1\}^n \to \{0,1\}$ is not exceptional. If $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ is a public key encryption scheme that supports efficient strong homomorphic evaluation of f with negligible error, then $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ has an efficient strong rerandomization with negligible error.*

5.4 One-sided rerandomization from homomorphic evaluation

In this section we show how to convert a homomorphic evaluation algorithm for a sensitive function into a one-sided rerandomization. In Section 5.8 we extend these ideas to obtain stronger notions of rerandomization under stronger assumptions. Let H denote entropy and I denote mutual information.

Claim 5.7. *Let $X_1, \ldots, X_n$ be i.i.d. random variables and $I \sim \{1, \ldots, n\}$ a uniformly random index. Let F, G, G' be random variables such that (1) F and G are independent conditioned on X_I, (2) F is independent of I, (3) G and G' are identically distributed and (4) F and G' are independent. Then the random variables (F, G) and (F, G') are within statistical distance $\sqrt{2\,\mathrm{H}(F)/n}$.*

Proof.

$$\begin{aligned}&\mathrm{H}(X_I \mid F) \geq \mathrm{H}(X_I \mid F, I) \quad \text{(conditioning reduces entropy)}\\ &= \frac{1}{n}\sum_{i=1}^{n} \mathrm{H}(X_i \mid F) \geq \frac{1}{n}\,\mathrm{H}(X_1, \ldots, X_n \mid F) \geq \frac{1}{n}(\mathrm{H}(X_1, \ldots, X_n) - \mathrm{H}(F)) = \mathrm{H}(X_I) - \frac{\mathrm{H}(F)}{n}.\end{aligned}$$

Since F and G are conditionally independent of X_I, $\mathrm{I}(F; G) \leq \mathrm{I}(F; X_I)$. Therefore

$$\mathrm{I}(F; G) \leq \mathrm{I}(F; X_I) = \mathrm{H}(X_I) - \mathrm{H}(X_I \mid F) \leq \frac{\mathrm{H}(F)}{n}$$

and the conclusion follows by Pinsker's inequality [Pin64]. □

The following lemma shows how to obtain one-sided rerandomization from homomorphic evaluation of a sensitive function. This lemma will be used in the proofs of Theorems 5.4 and 5.5. In Section 5.8 we give a version of this lemma that applies to a more restricted class of functions but allows us to achieve a stronger notion of rerandomization. That version will be used for the proof of Theorem 5.6.

Lemma 5.8. *Assume f has certifiable polynomial b-sensitivity and let δ be any function inverse polynomial in the security parameter. If $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ has a weak efficient homomorphic evaluator for f with error ε, then it has a one-sided weak rerandomization $\mathbf{Rer}^b$ with decryption error ε and rerandomization error δ.*

Proof. Suppose f_k has b-sensitivity k^α. Choose $k = (2c/\delta^2)^{1/\alpha}$, where c is the length of ciphertexts (for the given security parameter). Let (x, S) be the witness for b-sensitivity of f_k. Given public key PK and ciphertext E define $\mathbf{Rer}^b$ as follows:

1. Choose a random $I \sim S$.
2. Let
$$X_i = \begin{cases} \mathbf{Enc}_{PK}(x_i, R_i) & \text{if } i \neq I, \\ E & \text{if } i = I. \end{cases}$$
3. Output $F = H_{PK}(X_1, \ldots, X_k)$.

We first condition on the choice of the public key PK, letting ε_{PK} denote the statistical distance between the two distributions in the definition of strong homomorphic evaluator conditioned on PK.

The decryption error of $\mathbf{Rer}^b$ follows directly from the definition. We now show the rerandomization error is at most δ. Let F, G be two independent instantiations of $\mathbf{Rer}^b$ on the same input E. Conditioned on PK, the random variables $X_i \colon i \in S$ and F, G satisfy the assumptions of Claim 5.7. It follows that (F, G) and (F, G'), where G' is i.i.d with G and therefore with F, are within statistical distance $\sqrt{2c/k^\alpha}$, which is at most δ by our choice of parameters. Averaging over ε_{PK} we prove the lemma. □

5.5 The distinguishing protocol

In this section we describe a constant-round interactive proof system DP that, given input (PK, C, b), certifies that C is an encryption of b under PK when $b \in \{0, 1\}$ and

that (PK, C) is an invalid pair when $b = \perp$. The proof system is parametrized by two gaps $[\ell, r)$ and $[\ell', r')$, which describe a promise on the inputs.

We will assume we have the following constant-round protocols for statistical closeness ($SC_{[\ell,r)}$) and statistical farness ($SF_{[\ell,r)}$), where $0 \leq \ell < r \leq 1$. The protocols take as inputs a pair of sampler circuits D, D' producing distributions over the same set $\{0,1\}^m$ with the following completeness / soundness properties:

- If D, D' are at statistical distance less than ℓ, $SC_{[\ell,r)}(D, D')$ accepts with probability $1 - \sigma$.
- If D, D' are at statistical distance at least r, $SC_{[\ell,r)}(D, D')$ rejects with probability $1 - \sigma$.
- If D, D' are at statistical distance at least r, $SF_{[\ell,r)}(D, D')$ accepts with probability $1 - \sigma$.
- If D, D' are at statistical distance less than ℓ, $SF_{[\ell,r)}(D, D')$ rejects with probability $1 - \sigma$.

Here σ can be any inverse polynomial in the size of the input. The following two theorems state the existence of these protocols. The second one is stronger as it provides statistical zero-knowledge implementation, but makes a stronger assumption about the gaps.

Formally we will view SC and SF as promise problems that take ℓ, r, D, D' as their inputs. Theorem 5.9 essentially follows from work of Bhatnagar, Bogdanov, and Mossel [BBM11]. We provide the missing details in Appendix C.

Theorem 5.9. *For $r > \ell$, the problems SC and SF are in* AM *where the running time of the verifier is polynomial in the size of D, the size of D', and $1/(r - \ell)$.*

Theorem 5.10 is proved by Sahai and Vadhan [SV03].

Theorem 5.10. *For $r^2 > \ell$, the problems SC and SF are in* SZK *where the running time of the verifier is polynomial in the size of D, the size of D', and $1/\ell^{1/\log(r^2/\ell)}$.*

The protocol DP will certify that the rerandomization of C is close to an rerandomized encryption of b but far from a rerandomized encryption of $\bar{b}$ when $b \in \{0,1\}$. When $b = \perp$, it certifies that either the rerandomized encryptions of 0 and 1 are close, or the rerandomized encryption of C is far from both.

For $b \in \{0,1\}$, let $Z_{PK,b}$ be the circuit that on input R, R' outputs $\mathbf{Rer}^b_{PK}(\mathbf{Enc}_{PK}(b, R), R')$, i.e. a one-sided rerandomized encryption of b.

The distinguishing protocol $DP_{[\ell,r),[\ell',r')}$

On input (PK, C, b), where $b \in \{0, 1, \perp\}$:

1. If $b = 0$ or $b = 1$:
2. Verifier and Prover execute $SF_{[\ell,r)}(Z_{PK,0}, Z_{PK,1})$.
3. If the protocol rejects, reject. Otherwise:
4. Verifier and Prover execute $SC_{[\ell',r')}(Z_{PK,b}, \mathbf{Rer}^b_{PK}(C))$.
5. If the protocol accepts, accept, else reject.
6. If $b = \perp$:
7. Verifier and Prover execute $SC_{[\ell,r)}(Z_{PK,0}, Z_{PK,1})$.
8. If the protocol accepts, accept. Otherwise:
9. Verifier and Prover execute $SF_{[\ell',r')}(Z_{PK,0}, \mathbf{Rer}^0_{PK}(C))$.
10. Verifier and Prover execute $SF_{[\ell',r')}(Z_{PK,1}, \mathbf{Rer}^1_{PK}(C))$.
11. If both accept, accept, else reject.

The distinguishing oracle We now define an oracle π that distinguishes between encryptions of 0 and encryptions of 1. This oracle answers $\perp$ on all queries (PK, C) that do not represent valid key-ciphertext pairs and answers bad on all queries that fall inside the gaps of the underlying protocols SC and SF. We then show that for every pair (PK, C) that falls outside the gaps, $b = \pi(PK, C)$ is the unique answer that makes $DP(PK, C, b)$ accept.

Assume $\ell' < r/2$ and consider the following oracle

$$\pi_{[\ell,r),[\ell',r')}(PK, C) = \begin{cases} \perp, & \text{if } d < \ell \text{ or } (d \geq r \text{ and } d_0 \geq r' \text{ and } d_1 \geq r') \\ 0, & \text{if } d \geq r \text{ and } d_0 < \ell' \text{ (and so } d_1 \geq \ell') \\ 1, & \text{if } d \geq r \text{ and } d_1 < \ell' \text{ (and so } d_0 \geq \ell') \\ \text{bad}, & \text{if } d \in [\ell, r) \text{ or } d_0 \in [\ell', r') \text{ or } d_1 \in [\ell', r') \end{cases}$$

where $d = \mathrm{sd}(Z_{PK,0}, Z_{PK,1})$ and $d_b = \mathrm{sd}(Z_{PK,b}, \mathbf{Rer}^b_{PK}(C))$ (for $b \in \{0, 1\}$).

Let $\pi = \pi_{[\ell,r),[\ell',r')}$ and $DP = DP_{[\ell,r),[\ell',r')}$. The following claim shows that π is a distinguishing oracle.

Claim 5.11. *Assume* $\mathbf{Rer}^0, \mathbf{Rer}^1$ *are one-sided rerandomizations with decryption error* $\varepsilon < (1 - r)^2/2$ *and rerandomization error* $\rho < \ell'^2$. *Then* $\Pr[\pi(PK, \mathbf{Enc}_{PK}(b)) = b] \geq 1 - \sqrt{2\varepsilon} - \sqrt{\rho}$ *for every* $b \in \{0, 1\}$.

Proof. First we show that the statistical distance between the distributions

$$(PK, \mathbf{Rer}^0_{PK}(\mathbf{Enc}_{PK}(0,R), R') = Z_{PK,0}) \quad \text{and} \quad (PK, \mathbf{Rer}^1_{PK}(\mathbf{Enc}_{PK}(1,R), R') = Z_{PK,1})$$

is at least $1-2\varepsilon$. Consider the test T that on input (PK, C), samples SK conditioned on PK, and outputs $\mathbf{Dec}_{SK}(C)$. Since $\mathbf{Rer}^b$ are one-sided rerandomizations with decryption error ε, we have for every $b \in \{0,1\}$

$$\Pr[\mathbf{Dec}_{SK}(\mathbf{Rer}^b_{PK}(\mathbf{Enc}_{PK}(b))) = b] \geq 1-\varepsilon.$$

Therefore T distinguishes the two distributions with advantage $1-2\varepsilon$. By Markov's inequality, for at least a $1-\sqrt{2\varepsilon}$ fraction of the PK, the statistical distance between $Z_{PK,0}$ and $Z_{PK,1}$ is at least $1-\sqrt{2\varepsilon}$. Since $\mathbf{Rer}^b$ has a rerandomization error ρ, the statistical distance between

$$(PK, \mathbf{Rer}^b_{PK}(C,R), \mathbf{Rer}^b_{PK}(C,R')) \quad \text{and} \quad (PK, \mathbf{Rer}^b_{PK}(C,R), \mathbf{Rer}^b_{PK}(C',R'))$$

(where $C, C' \sim \mathbf{Enc}_{PK}(b)$ are independent) is at most ρ. By Markov's inequality, for at least a $1-\sqrt{\rho}$ fraction of the pairs (PK, C), the statistical distance between $\mathbf{Rer}^b_{PK}(C,R')$ and $\mathbf{Rer}^b_{PK}(C',R') = Z_{PK,b}$ is at most $\sqrt{\rho} < \ell'$. The claim follows by taking a union bound. □

The following claims are immediate from the definitions and the completeness and soundness assumptions on SC and SF.

Claim 5.12. *(Completeness) Assume $\ell' < r/2$ and $\pi(PK, C) \neq \text{bad}$. Then $DP(PK, C, \pi(PK, C))$ accepts with probability at least $1-3\sigma$.*

Claim 5.13. *(Soundness) Assume $\ell' < r/2$. If $DP(PK, C, b)$ accepts with probability more than 3σ, then $\pi(PK, C) \in \{b, \text{bad}\}$.*

5.6 Complexity theoretic setup

In this section we cover the complexity-theoretic framework for the proofs of Theorems 5.4 and 5.5.

5.6.1 Promise oracles for adaptive reductions

Let Ξ be any finite set of values that includes the special symbol bad. An *oracle family* over input length m with size d is a multiset Π of functions $\pi\colon \{0,1\}^m \to \Xi$. We say Π

is ε-*bad* if for every input x, $\Pr_{\pi\sim\Pi}[\pi(x) = \text{bad}] \leq \varepsilon$.

Let $F\colon \{0,1\}^m \to [d]$ be a function. The oracle $\Pi_F\colon \{0,1\}^m \to \Xi$ is given by $\Pi_F(z) = \pi_{F(z)}(z)$. In the lemma below F will be a randomized function of the same form.

Lemma 5.14. *Let $R^?$ be a reduction that on an input of length n, makes at most q queries of length m. Let Π be an oracle family of size d. Assume d is a power of two. There exists a randomized function $F\colon \{0,1\}^m \to [d]$ such that:*

- *F is computable in time (and hence uses randomness) polynomial in m, q, and d.*
- *For every input x of length n, the probability that $R^{\Pi_F}(x)$ never receives* bad *as an answer to any of its queries is at least $(1-\varepsilon)^q$.*

Proof. Fix m and let $F\colon \{0,1\}^m \to [d]$ be a q-wise independent function family. Using standard constructions, F can be described by $O(mq)$ random bits and is computable in time polynomial in m, q, and d.

Let $(Q_1, a_1), \ldots, (Q_q, a_q)$ denote the query-answer pairs of the reduction when interacting with the oracle Π_F. We may assume all queries are distinct. We write the probability that any of the a_i's equals bad as a product of conditional probabilities. Let p_i be the probability that $a_i \neq$ bad conditioned on $a_1, \ldots, a_{i-1} \neq$ bad.

Let's look at p_1 first. Since Π is ε-bad, the probability that a_1 is bad is at most ε and $p_1 \geq 1-\varepsilon$. Now we consider p_i. Since F is q-wise independent it follows that conditioned on every possible collection of values of $F(Q_1), \ldots, F(Q_{i-1})$ (which in particular determine the event $a_1, \ldots, a_{i-1} \neq$ bad), $F(Q_i)$ is uniformly distributed in $[d]$. Since Π is ε-bad, the conditional probability that $a_i =$ bad can be at most ε, and so $p_i \geq 1-\varepsilon$. Multiplying the conditional probabilities gives the second part of the lemma. □

5.6.2 Statistical zero-knowledge

We recall some results about the complexity of statistical zero-knowledge SZK. Sahai and Vadhan [SV03] show that the statistical distance problem $SD = SF_{[1/9,8/9)}$ is complete for SZK under many-one reductions.

We also need the following result of Sahai and Vadhan [SV03] that states the closure of SZK under truth-table reductions.

Theorem 5.15. *There is a deterministic algorithm that takes as input instances $x_1, \ldots, x_k$ of SD and a boolean predicate $P\colon \{0,1\}^k \to \{0,1\}$ and outputs an instance x of SD such that $SD(x) = P(SD(x_1), \ldots, SD(x_k))$. The running time of the algorithm is polynomial in 2^k and the sizes of $x_1, \ldots, x_k$.*

We also need the following fact, which says that reductions within SZK can without loss of generality be assumed deterministic.

Claim 5.16. *If there is a randomized many-one reduction R from L to SD such that $\Pr[SD(R(x)) = L(x)] \geq p$, where p is any constant above $1/2$, then L is in* SZK.

Proof. The reduction takes input x and randomness r and produces a pair of circuits D, D'. Let $E_x(r, s)$ (resp. $E'_x(r, s)$) be the circuits that on input r, s runs the reduction on input x and randomness r and outputs $D(s)$ (resp., $D'(s)$).

Assume $L(x) = SD(R(x))$ with probability at least $8/9$ over the randomness of the reduction. For $x \in L$, the statistical distance between E_x and E'_x is at least $(8/9)^2 \geq 2/3$ because at least $8/9$ choices of r contribute at least $8/9$ to the statistical distance. If $x \notin L$, then the statistical distance is at most $8/9 \cdot 1/9 + 1/9 \cdot 1 \leq 1/3$, because for at least $8/9$ choices of r the statistical distance over s is at most $1/9$, and for the other choices it is at most 1. Therefore L reduces deterministically to $SF_{[1/3,2/3)}$, so L is in SZK by Theorem 5.10.

If $\Pr[L(x) = SD(R(x))]$ is any constant above $1/2$, the probability can be first amplified to $8/9$ via Theorem 5.15 with the majority predicate. □

Combining Theorem 5.15 and Claim 5.16 we get the following corollary.

Corollary 5.17. *Suppose there is a randomized algorithm A that on input x of length n and randomness r computes inputs $x_1, \ldots, x_k$ and a predicate $P\colon \{0,1\}^k \to \{0,1\}$, where $k = O(\log n)$ and accepts if $P(SD(x_1), \ldots, SD(x_k))$ is true. If $\Pr[A(x) = L(x)] \geq p$, where p is any constant greater than $1/2$, then L is in* SZK.

5.7 Proofs of the main theorems

5.7.1 Proof of Theorem 5.4

Let $F_\omega\colon \{0,1\}^m \to [d]$ be the randomized function from Lemma 5.14, with ω describing the randomness. We set $I_j = \left[\frac{1}{3} + \frac{j-1}{3d}, \frac{1}{3} + \frac{j}{3d}\right)$ and $I'_j = \frac{1}{3}I_j$, where $1 \leq j \leq d$.

The decision protocol DL: On input x:

V: Compute the oracle query length m. Let d be the smallest power of two above $90q$ where q is an upper bound on the number of queries $R^?(x)$ makes. Choose randomness r for the reduction and randomness ω for F_ω. Send r, d, ω to the prover.

P: Send a sequence $((PK_i, C_i), b_i)$, $1 \leq i \leq q$ of oracle query-answer pairs.

V: Verify that the received query-answer pairs determine an accepting computation of $R^?(x,r)$. If not, reject. For every query i, compute $j = F_\omega(PK_i, C_i)$ and let $[\ell_i, r_i) = I_j$ and $[\ell'_i, r'_i) = I'_j$.

V, P: Execute in parallel the protocols $DP_{[\ell_i,r_i),[\ell'_i,r'_i)}(PK_i, C_i, b_i)$ for $1 \le i \le q$ with completeness/soundness gap $\sigma = 1/9q$. If any of them result in rejection, reject. Otherwise, accept.

Let $\pi_j = \pi_{I_j, I'_j}$ and Π_F be the oracle from Lemma 5.14.

Claim 5.18. *The oracle family $\{\pi_j\}_{1\le j\le d}$ is at most $3/d$-bad.*

Proof. Query (PK, C) is bad for π_j if $\mathrm{sd}(Z_{PK,0}, Z_{PK,1}) \in I_j$ or $\mathrm{sd}(Z_{PK,0}, \mathbf{Rer}_{PK}(C)) \in I'_j$ or $\mathrm{sd}(Z_{PK,1}, \mathbf{Rer}_{PK}(C)) \in I'_j$. Since the intervals I_j are disjoint, and so are the intervals I'_j, each of the three events occurs with probability at most $1/d$, so their union occurs with probability at most $3/d$. □

Proof of Theorem 5.4 It is sufficient to prove that $L \in \mathrm{AM}$. By applying the same argument to its complement $\overline{L}$ we also get $L \in \mathrm{coAM}$.

Assume $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ supports homomorphic evaluation of f with error at most ε and there is a reduction $R^?$ from L to $(1-\delta)$-distinguishing encryptions.

We instantiate the constructions with the following parameters. Let ε be the homomorphic evaluation error and c an upper bound on the length of ciphertexts queried by the reduction on input x. Let $\mathbf{Rer}^b$ be the rerandomization from Lemma 5.8 with parameters chosen so that the decryption error is ε and the rerandomization error is at most $\rho \le \varepsilon^2$. The protocol DP is instantiated with the rerandomizations $\mathbf{Rer}^0$ and $\mathbf{Rer}^1$.

Claim 5.19. *For an appropriate choice of parameters and for every F, Π_F is a $(1-\delta)$-distinguishing oracle.*

Proof. Notice that all the intervals I_j are within $[1/3, 2/3)$ and I'_j are all within $[1/9, 2/9)$. Since $\varepsilon < 1/18$ and $\rho < 1/81$, we have for every j, π_j satisfies the assumptions of Claim 5.11, which shows that

$$\Pr[\pi_j(PK, \mathbf{Enc}_{PK}(b)) = b] \ge 1 - \sqrt{2\varepsilon} - \sqrt{\rho} \ge 1 - 2\sqrt{\varepsilon} \ge 1 - \delta.$$

Since Π_F equals some π_j on every query, it follows that the same formula is true for Π_F, so Π_F is a $(1-\delta)$-distinguishing oracle. □

By Theorem 5.9, the verifier for DL can be implemented in polynomial time. Theorem 5.4 the follows by the next two claims:

Claim 5.20. *(Completeness) If $x \in L$, there exists a prover that makes $DL(x)$ accept with probability at least $2/3$.*

Proof. In the second step, the prover will emulate $R^{\Pi_F}(x, r)$. If the oracle returns bad on any of the queries in this emulation, the prover aborts (causing the verifier to reject). In the fourth step, the prover emulates the honest prover for $DP_{[\ell_i, r_i), [\ell'_i, r'_i)}$.

Let B be the event that $R^{\Pi_F}(x, r)$ rejects or Π_F returns bad on any of the queries in $R^{\Pi_F}(x, r)$ or any of DP protocols rejects. If B does not happen, then the verifier accepts. We upper bound the rejecting probability of the verifier by taking a union bound. Since Π_F is a distinguishing oracle, R^{Π_F} rejects with probability at most $1/9$. By Claim 5.18 and Lemma 5.14, R^{Π_F} returns bad with probability at most $1 - (1 - 3/d)^q \leq 1/9$; by Claim 5.12, each of the step 4 protocols rejects with probability at most $1/9q$, so by a union bound B happens with probability at most $1/3$. □

Claim 5.21. *(Soundness) If $x \notin L$ then no prover makes $DL(x)$ accept with probability at least $1/3$.*

Proof. Assume $x \notin L$. If $DL(x)$ accepts, then at least one of the following must be true:

1. $R^{\Pi_F}(x, r)$ accepts, or
2. Π_F returns bad on at least one query in $R^{\Pi_F}(x, r)$, or
3. $DL(x)$ accepts, $R^{\Pi_F}(x, r)$ rejects, and Π_F never returns bad.

We upper bound the probabilities of each of these events. Since Π_F is a distinguishing oracle, the first one occurs with probability at most $1/9$. By Claim 5.18 and Lemma 5.14, the second one occurs with probability at most $1 - (1 - 3/d)^q \leq 1/9$. If the third event is satisfied, then b_i must differ from $\Pi_F(PK_i, C_i) = \pi_{p_i, q_i}(PK_i, C_i)$ for at least one i. By Claim 5.13, the i'th instantiation of the DP protocol in then accepts with probability at most $1/9$. By a union bound, $DL(x)$ accepts with probability at most $1/3$. □

5.7.2 Proof of Theorem 5.5

Let $I_j, 1 \leq j \leq d$ be the following collection of intervals: $I_j = [\ell_j, r_j)$ where $r_1 = 1/2$, $\ell_j = r_j^2/4$, and $r_{j+1} = \ell_j$. Let $I'_j = \frac{1}{3} I_j$. Assume the reduction makes at most q queries on every input and let $d = 27q \cdot 3^q$.

By Theorem 5.10, for every j the problems $SC_{I_j}, SC_{I'_j}, SF_{I_j}, SF_{I'_j}$ are all in SZK so by Theorem 5.15 and the completeness of SD, DP_{I_j,I'_j} is also in SZK for every j.

Consider the following algorithm A. On input x, choose randomness r for R and a random $j \sim [d]$ and accept if there exists a sequence of answers $(a_1, \ldots, a_q) \in \{0, 1, \perp\}^q$ such that $R(x, r)$ accepts given these oracle answers and $DP_{I_j,I'_j}(Q_i, a_i)$ accepts for all $1 \leq i \leq q$. Since DP_{I_j,I'_j} is in SZK and SD is complete for SZK, A satisfies the assumption of Corollary 5.17, so if we can prove that $\Pr[A(x) = L(x)] \geq 2/3$, it will follow that L is in SZK.

Say j is bad if $\pi_j = \pi_{I_j,I'_j}$ answers bad on any pair (Q, a) queried by A. Since A makes at most $q3^q$ queries, by Claim 5.18 and a union bound the probability that A answers bad on any of its queries is at most $1/9$.

Fix an input x. By our choice of parameters, when ε is sufficiently small and $\rho = \varepsilon^2$, Claim 5.11 guarantees that π_j is a $(1 - 4\varepsilon)$-decryption oracle for every $1 \leq j \leq d$. So for at least $8/9$ fraction of r, $R^{\pi_j}(x, r) = L(x)$. Therefore with probability at least $7/9$, both $R^{\pi_j}(x, r) = L(x)$ and π_j never answers bad on any of A's queries. By Claims 5.12 and Claim 5.13, it must then hold that $a = \pi_j(Q)$ for all query-answer pairs (Q, a) made by A, and so $A(x) = L(x)$.

5.8 Strong rerandomization from strong homomorphic evaluation

In this Section we prove Theorem 5.6. We begin by defining "t-symmetric functions".

t-symmetric functions Let G be a subgroup of the symmetric group S_k and $x \in \{0, 1, \star\}^k$ be a string containing exactly one $\star$. Let $t_0(G, x)$ (resp., $t_1(G, x)$) be the number of transpositions $\tau \in G$ that transpose a 0 and a $\star$ (resp., a 1 and a $\star$) when acting on x. Observe that $t_b(G, \sigma x) = t_b(G, x)$ for every $\sigma \in G$.

Let $x|_{\star\to 0}, x|_{\star\to 1}$ be the string obtained when the $\star$ in x is replaced by a 0 and a 1 respectively. We will say a boolean function $f\colon \{0,1\}^k \to \{0,1\}$ is t-symmetric if there exist x and G with $t_0(G, x), t_1(G, x) > t$ and $f(\sigma x|_{\star\to b}) = b$ for every $\sigma \in G$.

For example, the majority function on 3 bits is 2-symmetric: Take $G = S_3$ and let $x = 01\star$. So is parity on 4 bits: Take $G = S_4$ and $x = 110\star$. The DNF $(x_{11} \wedge x_{12}) \vee (x_{21} \wedge x_{22})$ is also 2-symmetric. To see this take x to be the string $x_{11} = \star, x_{12} = 1, x_{21} = 0, x_{22} = 1$ and G to be the "wreath product" $S_2 \wr S_2$, which acts on x by first permuting the inputs in each term of the DNF independently, then permuting the terms.

Proof of Theorem 5.6 The theorem follows from the next two claims, proved below.

Claim 5.22. *Let $f\colon \{0,1\}^k \to \{0,1\}$, $k \geq 2$ be any boolean function that depends on all its inputs and is not one of OR / AND. If $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ supports efficient strong homomorphic evaluation of f with error ε, then $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ supports efficient strong homomorphic evaluation of a 2-symmetric function with error at most 12ε.*

Claim 5.23. *Let $f\colon \{0,1\}^k \to \{0,1\}$ be a 2-symmetric function. If $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ is a public key encryption scheme that supports efficient strong homomorphic evaluation of f with negligible error, then $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ has an efficient strong rerandomization with negligible error.*

5.8.1 Proof of Claim 5.22

Claim 5.24. *Let $f\colon \{0,1\}^k \to \{0,1\}$, $k \geq 2$ be a monotone function that depends on all its inputs.*

1. *If f is not the AND function, then f has 0-sensitivity at least 2.*
2. *If f is not the OR function, then f has 1-sensitivity at least 2.*

Proof. We prove (1) by induction on k, the proof of (2) is analogous. The base case $k = 2$ follows by inspection.

For the inductive step, let f_0 and f_1 be f with x_1 fixed to 0 and 1 respectively. If one of f_0 or f_1 is 2-sensitive, we are done. Otherwise, by inductive hypothesis, both f_0 and f_1 are AND functions of their relevant variables. Let S_0 and S_1 be the set of these relevant variables respectively.

Since f is monotone, either S_0 is empty or S_0 contains all of S_1. Also note that $S_0 \cup S_1 = \{2, \ldots, k\}$ because f depends on all its inputs. If S_0 is empty, then $S_1 = \{2, \ldots, k\}$ and so f is the AND function. Otherwise, $S_0 = \{2, \ldots, k\}$.

If S_1 also equals $\{2, \ldots, k\}$, then f does not depend on x_1. Otherwise, there exists at least one variable that is in S_0 but not in S_1. Without loss of generality let this variable be x_2. In this case the pair $(0011...1, \{1, 2\})$ witnesses the 2-sensitivity of f. □

Let $f\colon \{0,1\}^k \to \{0,1\}$ be a boolean function. We say f is an *extension* of g if there exists a set $S \in [k]$ and $z \in \{0,1\}^{\overline{S}}$ such that g is the restriction of f to S using z, i.e. $f_{S|z}(x) = g(x)$ for every $x \in \{0,1\}^S$.

We omit the proof of the following facts.

Claim 5.25. *Let g be a function with b-sensitivity at least s and f be any extension of g. Let $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ be a public key encryption scheme. If $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ supports strong (resp. weak) homomorphic evaluation of f with error ε, $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ supports strong (resp. weak) homomorphic evaluation of g with error ε.*

Claim 5.26. *Let $g\colon \{0,1\}^k \to \{0,1\}$ be a boolean function. For every $i \in [k]$, let $f_i\colon \{0,1\}^{k_i} \to \{0,1\}$ be a boolean function. Let $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ be a public key encryption scheme. If $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ supports strong homomorphic evaluation of g with error ε and each of the f_i's with error ε_i, then $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ supports strong homomorphic evaluation of $g(f_1, \ldots, f_k)$ with error $\varepsilon + \varepsilon_1 + \cdots + \varepsilon_k$.*

Proof of Claim 5.22. First, we show that $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ supports homomorphic evaluation of f_0 and f_1 with error at most 4ε, where f_b has b-sensitivity 2. Consider the 2-symmetric function $g\colon \{0,1\}^4 \to \{0,1\}$ defined by $g(x_{11}, x_{12}, x_{21}, x_{22}) = f_0(f_1(x_{11}, x_{12}), f_1(x_{21}, x_{22}))$. Since g is a composition of f_0 and f_1, by Claim 5.26 $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ has a strong homomorphic evaluation of g with error at most 12ε.

Now we show that $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ supports homomorphic evaluation of f_0 and f_1. This follows from Claim 5.24 and 5.25 if f is monotone. If f is not monotone, there is an $x \in \{0,1\}^k$ and $i \in [k]$ such that $x_i = 1$, $f(x) = 0$ and $f(x|_i) = 1$. Let g be the restriction of f to the rest of the bits using x_i. Note that g is the NOT function and so by Claim 5.25 $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ supports homomorphic evaluation of the NOT function with error ε. It is easy to see that one can obtain f_0 and f_1 by composing g with a restriction of f. The rest follows by Claim 5.26. □

5.8.2 Proof of Claim 5.23

We start with the following Corollary of Claim 5.7 for the special case when $G = X_I$.

Corollary 5.27. *Let $X_1, \ldots, X_n$ be i.i.d and $I \sim \{1, \ldots, n\}$ a uniformly random index and F be independent of I. Then (F, X_I) and (F, X) are within statistical distance $\sqrt{2\,\mathrm{H}(F)/n}$, where X is i.d. with $X_1, \ldots, X_n$ and independent of F.*

The following lemma shows how to obtain strong rerandomization from any t-symmetric function. The resulting rerandomization error is noticeable.

Lemma 5.28. *Let $f\colon \{0,1\}^k \to \{0,1\}$ be any t-symmetric function. If $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ has a strong efficient homomorphic evaluator for f with error ε, then it has a strong efficient rerandomization* **Rer** *with error at most $\varepsilon + \sqrt{2c/t}$ (resp. decryption error ε and rerandomization error $\sqrt{2c/t}$), where c is the length of ciphertexts.*

Proof. Define **Rer** to be the following procedure. Let $x \in \{0,1,\star\}^k$ and G be the subgroup of S_k that witness the t-symmetry of f. Given public key PK and ciphertext E we define **Rer** as follows:

1. Let
$$X_i = \begin{cases} \mathbf{Enc}_{PK}(x_i, R_i) & \text{if } x_i \in \{0,1\}, \\ E & \text{if } x_i = \star. \end{cases}$$
2. Choose a random permutation π from G.
3. Output $F = H_{PK}(X_{\pi(1)}, \ldots, X_{\pi(k)})$.

We will now assume that $E \sim \mathbf{Enc}_{PK}(0)$; the case $E \sim \mathbf{Enc}_{PK}(1)$ is similar. We first condition on the choice of the public key PK, letting ε_{PK} denote the statistical distance between the two distributions in the definition of strong homomorphic evaluator conditioned on PK.

Let $S \subseteq [k]$ be the set of indices i such that $x_i = 0$ and there is a transposition π in G which swaps i and the $\star$ coordinate of x. By assumption $|S| \geq t$. The random variables $X_i \colon i \in S$ and F satisfy the assumptions of the Corollary 5.27, so (X_I, F) (where $I \sim S$) is within statistical distance $\sqrt{2c/t}$ from (E', F), where $E' \sim \mathbf{Enc}_{PK}(0)$ is independent of F. On the one hand, by the randomness of π, (X_I, F) is identically distributed to $(E, F) = (E, \mathbf{Rer}_{PK}(E))$. On the other hand, by the strong homomorphic property, (E', F) is within distance ε_{PK} of a pair of independent random encryptions of 0 under PK. So conditioned on PK, the statistical distance in rerandomization is at most $\varepsilon_{PK} + \sqrt{2c/t}$. Averaging over ε_{PK} we prove the strong version of the lemma. □

We now show that for strong homomorphic evaluation, the error can be reduced and prove Theorem 5.6.

For a boolean function $f \colon \{0,1\}^k \to \{0,1\}$, Let $f^{(r)} \colon \{0,1\}^{k^r} \to \{0,1\}$ be defined recursively by first applying $f^{(r-1)}$ on k independent tuples of k^{r-1} inputs and then applying f to these k values. For the base case we take $f^{(1)} = f$.

Claim 5.29. *If f is t-symmetric, then $f^{(r)}$ is t^r-symmetric.*

Proof. Let x and G witness the t-symmetry of f. We construct $x^{(r)}$ and $G^{(r)}$ which show that $f^{(r)}$ is t^r symmetric:

- Let $x^{(1)} = x$ and obtain $x^{(r)}$ from $x^{(r-1)}$ by replacing every $\star$ in $x^{(r-1)}$ by x, every 0 by $x|_{\star \to 0}$ and every 1 by $x|_{\star \to 1}$.

- Let $G^{(1)} = G$ and $G^{(r)}$ be the wreath product $G \wr G^{(r-1)}$. The permutations in G are obtained by applying any collection of permutations of $G^{(r-1)}$ to each one of the r inputs to $f^{(r-1)}$ and then applying a permutation of G to the outputs of the k copies of $f^{(r-1)}$.

It is immediate from the construction that $t_0(G^{(r)}, x^{(r)}) = t_0(G, x)^r$, $t_1(G^{(r)}, x^{(r)}) = t_1(G, x)^r$, and $f(\sigma^{(r)} x^{(r)}|_{\star \to b}) = b$ for every $\sigma^{(r)} \in G^{(r)}$. □

Proof of Claim 5.23. Let **Rer** be the rerandomization of f from the proof of Lemma 5.28. We define $\mathbf{Rer}^{(r)}$ recursively by $\mathbf{Rer}^{(1)} = \mathbf{Rer}$ and

$$\mathbf{Rer}^{(r)}_{PK}(E, (R_1, \ldots, R_r)) = \mathbf{Rer}_{PK}(\mathbf{Rer}^{(r-1)}_{PK}(E, (R_1, \ldots, R_{r-1})), R_r).$$

where $R_1, \ldots, R_r$ are independent random strings. We now argue that $\mathbf{Rer}^{(r)}$ has the desired properties.

Let $\mathbf{Rer}'^{(r)}$ be the rerandomization obtained by applying the construction of Lemma 5.28 to the function $f^{(r)}$. We claim that the distributions $(PK, E, \mathbf{Rer}^{(r)}_{PK}(E))$ and $(PK, E, \mathbf{Rer}'^{(r)}_{PK}(E))$, where $E \sim \mathbf{Enc}_{PK}(b)$, are within statistical distance at most εk^{r-1}. We show this by induction. The base case $r = 1$ is obvious (the statistical distance is zero).

For the inductive step, we can describe $\mathbf{Rer}^{(r)}_{PK}(E)$ as follows: First, choose X by applying a random permutation π to the indices of $x \in \{0, 1, \star\}$. Then $\mathbf{Rer}^{(r)}_{PK}(E) = H_{PK}(e_1, \ldots, e_k)$ where

$$e_i = \begin{cases} \mathbf{Enc}_{PK}(X_i) & \text{when } X_i \neq \star \\ \mathbf{Rer}^{(r-1)}_{PK}(E) & \text{when } X_i = \star. \end{cases}$$

On the other hand $\mathbf{Rer}'^{(r)}_{PK}(E)$ can be described as follows: First, choose X by applying a random permutation π to the indices of $x \in \{0, 1, \star\}$. Then $\mathbf{Rer}'^{(r)}_{PK}(E) = H_{PK}(e'_1, \ldots, e'_k)$ where

$$e'_i = \begin{cases} \mathbf{Rer}'^{(r-1)}_{PK}(\mathbf{Enc}_{PK}(X_i)) & \text{when } X_i \neq \star \\ \mathbf{Rer}'^{(r-1)}_{PK}(E) & \text{when } X_i = \star. \end{cases}$$

By inductive assumption, the statistical distance between $(PK, \mathbf{Rer}^{(r-1)}_{PK}(E))$ and $(PK, \mathbf{Rer}'^{(r-1)}_{PK}(E))$ is at most εk^{r-2}. Since H_{PK} has error ε, the statistical distance between $(PK, \mathbf{Enc}_{PK}(b))$ and $(PK, \mathbf{Rer}'^{(r-1)}_{PK}(\mathbf{Enc}_{PK}(b))$ can also be bounded by εk^{r-2} using an inductive argument. Applying a hybrid argument we conclude that the distributions

$(PK, e_1, \ldots, e_k)$ and $(PK, e'_1, \ldots, e'_k)$ are within distance at most εk^{r-1} and therefore so are the distributions $(PK, \mathbf{Rer}^{(r)}_{PK}(E))$ and $(PK, \mathbf{Rer}'^{(r)}_{PK}(E))$.

By Claim 5.29, $f^{(r)}$ is t^r symmetric. It follows from Claim 5.26 that the function $H^{(r)}_{PK}$ defined recursively by $H^{(1)}_{PK} = H_{PK}$ and $H^{(r)}_{PK} = H_{PK}(H^{(r-1)}_{PK}, \ldots, H^{(r-1)}_{PK})$ is a homomorphic evaluation of $f^{(r)}$ with error at most εk^r. By Lemma 5.28, $\mathbf{Rer}'^{(r)}$ has error $k^r\varepsilon + \sqrt{2c/t^r}$. Therefore $\mathbf{Rer}^{(r)}$ has error at most $\varepsilon(k^{r-1} + k^r) + \sqrt{2c/t^r}$. Let $\alpha = \log t / \log k$. By choosing $r = 1/(2+\alpha) \cdot \log(2c/\varepsilon^2)/\log k$ we get that $\mathbf{Rer}^{(r)}$ has error $O(\varepsilon^{\alpha/(2+\alpha)})$, which is negligible when ε is negligible. □

Chapter 6

Conclusion

In this thesis, we have given partial answers to some complexity-theoretic questions concerning homomorphic encryptions. We proposed a code-based homomorphic encryption scheme, gave evidence that homomorphic encryption is inherently more complex than ordinary encryption schemes, and showed that its security cannot be proved beyond $\mathrm{AM} \cap \mathrm{coAM}$ under plausible complexity assumption. In this chapter we provide some relevant open questions whose answers may lead to deeper understanding of homomorphic encryption.

- Is code-based homomorphic encryption possible at all? can our ideas for constructing homomorphic encryption from codes be used elsewhere (e.g. other codes)? Also, all currently known fully homomorphic encryption schemes are based on lattices. Can we depart from lattices and construct one that is based on other assumptions?

- We showed that secure homomorphic encryption cannot be implemented in AC^0. Can we find complexity-theoretic evidence that homomorphic evaluation is somewhere above AC^0, e.g. it is NC^1-hard or L-hard?

- Can we extend our result on limits of provable security to other functionalities? Is there a public-key encryption scheme that supports homomorphic evaluation of the "insensitive" functions alone (e.g. only NOT)? Can we extend the result to show that its security cannot be proven beyond SZK, without requiring the reduction to have constant query complexity?

- We gave a method for converting homomorphic evaluation of any unexceptional functions to a rerandomization algorithm. What can we say about the exceptional

ones? Are there any other applications for ciphertext rerandomization?

Appendix A

The ranks of submatrices of the public key

We prove the following proposition, which points to the limitation of an attack on the public key of M described in the introduction.

Proposition A.1. *Let $T \subseteq [n]$, $|T| = t$ be an arbitrary subset of rows of the $r \times n$ public key matrix P such that $|T \cap S| \leq s/3 + \max\{t - r, 0\}$. Then the submatrix P_T of P spanned by the rows indexed by T has full rank with probability at least $1 - O(r^2/q)$, where the randomness is taken over the choice of $a_1, \ldots, a_n$ in the key generation algorithm.*

Proof. We prove the theorem for the matrix M instead of P. Since P and M have the same column space and the rank of P_T is a property of the column space of P projected to the coordinates in T, the statement will follow.

Without loss of generality we may assume that M_T is a square matrix: If $t < r$ we can augment the M_T by rows from outside S, and if $t > r$, we can eliminate rows from M_T that come from S (and some extra ones if necessary). Both operations preserve rank deficiency.

Now suppose M_T is a square matrix so that at most $s/3$ of its rows come from S. Let us assume, again without loss of generality, that $T = \{1, \ldots, r\}$ and $S = \{1, \ldots, s_0\}$, $s_0 \leq s/3$. We now argue that with probability $1 - O(r^2/q)$, the determinant $\det(M_T)$ is nonzero.

Notice that $\det(M_T)$ is a formal polynomial in the variables $a_1, \ldots, a_r$ of degree at most $1 + 2 + \cdots + r = r(r+1)/2$. In our setup, the diagonal term $a_1 a_2^2 \ldots a_r^r$ appears uniquely in the sum-product expansion of the determinant, and so this formal polynomial is nonzero. By the Schwarz-Zippel lemma, if $a_1, \ldots, a_r$ were chosen independently at

random from $\mathbb{F}_q$, $\det(M_T)$ would be zero with probability at most $1 - r(r+1)/2q$. Our a_i are not independent since they are required to be distinct, but the statistical distance between r uniformly independent elements of $\mathbb{F}_q$ and r uniform but distinct elements of $\mathbb{F}_q$ is only $O(r^2/q)$. It follows that $\det(M_T) \neq 0$ with probability $1 - O(r^2/q)$. □

Appendix B

Approximate $0,1$-majorities over arbitrary fields

In this section we prove the following claim.

Proposition B.1. *Let q be the power of a prime. There exists a circuit $APXMAJ_m \colon \mathbb{F}_q^m \to \mathbb{F}_q$ of size $O(m^2)$ and depth $O(\log m)$ with the property (3.3).*

The challenge is to make the depth of the circuit independent of q. We show an easy construction based on a trick of Valiant [Val84].

Proof. Let $CORR_d$ be the correction circuit from Section 3.4.1 where $d = 2\log m + 4$. We will show that there exists a way to connect the m inputs to the 2^d inputs of $CORR_d$ in a way that the resulting circuit computes $APXMAJ_m$.

Fix a specific input x so that at least $7/8$ of its elements equal b. If each of the inputs to $CORR_d$ is randomly wired to one of the elements in x, then the inputs to $CORR_d$ will take value b independently with probability at least $7/8$ each. Recall that for $b \in \{0,1\}$, if each of the inputs to this circuit takes value b with probability $7/8$, then its output takes value b with probability $1 - (3/4)^{2^{d/2}} > 1 - 2^{-m}$ by our choice of d. Taking a union bound over all such inputs x, we conclude that there must exist a wiring with the desired property. □

Appendix C

An AM protocol for statistical closeness

Bogdanov, Bhatnagar and Mossel [BBM11] show the existence of a protocol for statistical farness (SF) meeting the specifications of Theorem 5.9. They also give a protocol for statistical closeness (SC), but they only provide a soundness proof for gaps $[\ell, r)$ satisfying $r/\ell \geq 4$. We show how to extend their protocol and analysis to general gaps.

Theorem C.1. *For $r > \ell$, the problem $SC_{[\ell,r)}(D, D')$ is in* AM *where the running time of the verifier is polynomial in the size of D, the size of D', and $1/(r-l)$.*

Let $N(t) = |\{\omega : |D^{-1}(\omega)| \geq t \text{ and } |D'^{-1}(\omega)| \geq t\}|$. From [BBM11], there is a lower bound protocol for $N(t)$ with completeness $1 - \delta/20n$ and soundness $\delta/20n$. More specifically, they show that the following decision problem is in AM:

Input: A pair of circuits $D, D' : \{0,1\}^n \to \{0,1\}$, a number $1 \leq t \leq 2^n$, a target number $0 \leq \tilde{N} \leq 2^n$, and a fraction $0 < \delta \leq 1$ (represented in unary).
Yes instances: $(D, D', t, \tilde{N}, \delta)$ such that $N(t) \geq \tilde{N}$.
No instances: $(D, D', t, \tilde{N}, \delta)$ such that $N((1-\delta)t) < (1-\delta)\tilde{N}$.

Following the ideas of [BBM11] we have the following protocol for statistical closeness:

An AM **protocol for** SC**:** On input ℓ, r, D, D': Set $\delta = (r-\ell)/4$ and

P: Send claims $\tilde{N}_i$ for the values $N_i = N((1-\delta)^{-i}), 0 \leq i \leq en/\delta$.

P, V: Run the AM lower bound protocol for N_i on inputs $(D, D', (1-\delta)^{-i}, \tilde{N}_i, \delta)$ for every $1 \leq i \leq en/\delta$. If all of them pass accept, otherwise reject.

V: Accept if $\sum_{i=0}^{en/\delta}(\tilde{N}_i - \tilde{N}_{i+1})(1-\delta)^{-i} \geq (1-\delta)(1-\ell)\cdot 2^n$.

The completeness and soundness of the protocol rely on the following approximation from [BBM11]:

$$\sum_{i=0}^{en/\delta}(N_i - N_{i+1})(1-\delta)^{-i} \leq (1-\mathrm{sd}(D,D'))2^n \leq \sum_{i=0}^{en/\delta}(N_i - N_{i+1})(1-\delta)^{-(i+1)}.$$

Claim C.2 (Completeness)**.** *If* $\mathrm{sd}(D,D') \leq \ell$ *then the protocol accepts with probability* $2/3$.

Proof. Assume the honest prover claims that $\tilde{N}_i = N_i$ for every i. By the completeness of the lower bound protocol and a union bound, with probability at least $2/3$ none of the lower bound protocol rejects. In this case, using the above approximation we have

$$\sum_{i=0}^{en/\delta}(\tilde{N}_i - \tilde{N}_{i+1})(1-\delta)^{-i} \geq (1-\delta)(1-\mathrm{sd}(D,D'))\cdot 2^n. \qquad \square$$

Claim C.3 (Soundness)**.** *If the protocol accepts with probability at least* $1/3$ *then* $\mathrm{sd}(D,D') \leq r$.

Proof. Assume the verifier accepts with probability at least $1/3$. By the soundness of the lower bound protocol for $N(t)$ and a union bound, there exists at least one setting of the randomness of the verifier for which $N_{i-1} \geq (1-\delta)\tilde{N}_i$ for all i (where $N_{-1} = N_0$) and the verifier accepts. Now (using the fact that the value $N_{en/\delta+1}$ is zero):

$$\begin{aligned}
\sum_{i=-1}^{en/\delta}(N_i - N_{i+1})(1-\delta)^{-i} &= N_{-1}(1-\delta) + \sum_{i=0}^{en/\delta} N_i((1-\delta)^{-i} - (1-\delta)^{-i+1}) \\
&\geq \tilde{N}_0(1-\delta)^2 + (1-\delta)\sum_{i=0}^{en/\delta}\tilde{N}_{i+1}((1-\delta)^{-i} - (1-\delta)^{-i+1}) \\
&= (1-\delta)\cdot\sum_{i=0}^{en/\delta}(\tilde{N}_i - \tilde{N}_{i+1})(1-\delta)^{-i+1} \\
&\geq (1-\delta)^3(1-\ell)\cdot 2^n
\end{aligned}$$

so we get that $1-\mathrm{sd}(D,D') \geq 1-\ell$ and therefore $\mathrm{sd}(D,D') \leq 1-(1-\delta)^3(1-\ell) \leq \ell + 4\delta$. Setting $\delta = (r-\ell)/4$ proves the claim. $\square$

Bibliography

[AAPS11] Frederik Armknecht, Daniel Augot, Ludovic Perret, and Ahmad-Reza Sadeghi. On constructing homomorphic encryption schemes from coding theory. Technical Report 309, Cryptology ePrint Archive, 2011.

[ABW10] B. Applebaum, B. Barak, and A. Wigderson. Public-key cryptography from different assumptions. In *Proceedings of the 42th ACM Symposium on Theory of Computing*, pages 171–180, 2010.

[AGGM06] Adi Akavia, Oded Goldreich, Shafi Goldwasser, and Dana Moshkovitz. On basing one-way functions on NP-hardness. In *Proceedings of the 38th ACM Symposium on Theory of Computing*, 2006.

[AIK07] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. Cryptography with constant input locality. In *Proceedings of the 27th annual international cryptology conference on Advances in cryptology*, CRYPTO'07, pages 92–110, Berlin, Heidelberg, 2007. Springer-Verlag.

[Ajt96] M. Ajtai. Generating hard instances of lattice problems (extended abstract). In *Proceedings of the 28th ACM Symposium on Theory of Computing*, STOC '96, pages 99–108, New York, NY, USA, 1996. ACM.

[AS08] Frederik Armknecht and Ahmad-Reza Sadeghi. A New Approach for Algebraically Homomorphic Encryption. Technical Report 422, Cryptology ePrint Archive, 2008.

[BBM11] Nayantara Bhatnagar, Andrej Bogdanov, and Elchanan Mossel. The computational complexity of estimating convergence time. In *Proceedings of the 15th International Workshop on Randomization and Computation (RANDOM)*, 2011.

[BGN05] Dan Boneh, Eu-Jin Goh, and Kobbi Nissim. Evaluating 2-dnf formulas on ciphertexts. In Joe Killian, editor, *Proceedings of Theory of Cryptography Conference 2005*, volume 3378 of *LNCS*, pages 325–342. Springer, 2005.

[BGV12] Z. Brakerski, C. Gentry, and V. Vaikuntanathan. Fully Homomorphic Encryption without Bootstrapping. In *Innovations in Theoretical Computer Science*, 2012. To appear.

[BL11] Andrej Bogdanov and Chin Ho Lee. Homomorphic encryption from codes. Cryptology ePrint Archive, Report 2011/622, 2011.

[BL12] Andrej Bogdanov and Chin Ho Lee. On the depth complexity of homomorphic encryption schemes. Electronic Colloquium on Computational Complexity (ECCC), TR12-157, 2012.

[BL13] Andrej Bogdanov and Chin Ho Lee. Limits of provable security for homomorphic encryption. In *Proceedings of CRYPTO*, 2013. To appear.

[BLP+13] Zvika Brakerski, Adeline Langlois, Chris Peikert, Oded Regev, and Damien Stehlé. Classical hardness of learning with errors. In *Proceedings of the 45th annual ACM symposium on Symposium on theory of computing*, STOC '13, pages 575–584, New York, NY, USA, 2013. ACM.

[Bon98] Dan Boneh. The decision diffie-hellman problem. In *Algorithmic number theory*, pages 48–63. Springer, 1998.

[Bop97] Ravi B. Boppana. The average sensitivity of bounded-depth circuits. *Inf. Process. Lett.*, 63(5):257–261, September 1997.

[BOW10] Eric Blais, Ryan O'Donnell, and Karl Wimmer. Polynomial regression under arbitrary product distributions. *Machine Learning*, 80:273–294, 2010.

[Bra79] Gilles Brassard. Relativized cryptography. In *Proceedings of the 20th IEEE Symposium on Foundations of Computer Science*, pages 383–391, 1979.

[Bra13] Zvika Brakerski. When homomorphism becomes a liability. In *Proceedings of the Tenth Theory of Cryptography Conference*, 2013.

[BT06] Andrej Bogdanov and Luca Trevisan. On wost-case to average-case reductions for NP problems. *SIAM J. Comp.*, 36(4), 2006.

[BV11] Z. Brakerski and V. Vaikuntanathan. Efficient Fully Homomorphic Encryption from (Standard) LWE. In *Proceedings of the 53rd Annual Symposium on Foundations of Computer Science*, 2011.

[EY80] Shimon Even and Yacob Yacobi. Cryptography and NP-completeness. In *Proceedings of the 7th ICALP*, volume 85 of *LNCS*, pages 195–207. Springer-Verlag, 1980.

[FF93] Joan Feigenbaum and Lance Fortnow. Random-self-reducibility of complete sets. *SIAM Journal on Computing*, 22:994–1005, 1993.

[Gam85] T. El Gamal. A public-key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Trans. Info. Theory*, 31(4):469–472, 1985.

[Gen09a] Craig Gentry. *A fully homomorphic encryption scheme*. PhD thesis, Stanford University, 2009.

[Gen09b] Craig Gentry. Fully Homomorphic Encryption Using Ideal Lattices. In *STOC*, pages 169–178, 2009.

[GG98] Oded Goldreich and Shafi Goldwasser. On the possibility of basing cryptography on the assumption that $P \neq NP$. Unpublished manuscript, 1998.

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *J. ACM*, 33(4):792–807, August 1986.

[GH11] Craig Gentry and Shai Halevi. Fully Homomorphic Encryption without Squashing Using Depth-3 Arithmetic Circuits. In *Proceedings of the 53rd Annual Symposium on Foundations of Computer Science*, 2011.

[GL89] O. Goldreich and L. A. Levin. A hard-core predicate for all one-way functions. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, STOC '89, pages 25–32, New York, NY, USA, 1989. ACM.

[GM84] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *Journal of computer and system sciences*, 28(2):270–299, 1984.

[Gol00] Oded Goldreich. Candidate one-way functions based on expander graphs. *Electronic Colloquium on Computational Complexity (ECCC)*, 7(090), 2000.

[Gol04] O. Goldreich. *Foundations of Cryptography, vol. 2: Basic Applications*. Cambridge University Press, 2004.

[Gol08] O. Goldreich. *Computational Complexity: A Conceptual Perspective*. Cambridge University Press, 2008.

[GOT12] Valérie Gauthier, Ayoub Otmani, and Jean-Pierre Tillich. A distinguisher-based attack of a homomorphic encryption scheme relying on reed-solomon codes. Cryptology ePrint Archive, Report 2012/168, 2012.

[GRS08] Henri Gilbert, Matthew J. Robshaw, and Yannick Seurin. How to encrypt with the LPN problem. In *Proceedings of the 35th international colloquium on Automata, Languages and Programming, Part II*, ICALP '08, pages 679–690, Berlin, Heidelberg, 2008. Springer-Verlag.

[GT00] R. Gennaro and L. Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. In *Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, FOCS '00, pages 305–, Washington, DC, USA, 2000. IEEE Computer Society.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM J. Comput.*, 28(4):1364–1396, March 1999.

[HLW06] Shlomo Hoory, Nathan Linial, and Avi Wigderson. Expander graphs and their applications. *Bull. Amer. Math. Soc.*, 43:439–561, 2006.

[HMX10] I. Haitner, M. Mahmoody, and D. Xiao. A new sampling protocol and applications to basing cryptographic primitives on np. In *Proceeedings of 25th IEEE Conference on Computational Complexity (CCC)*, pages 76–87, 2010.

[HRV10] Iftach Haitner, Omer Reingold, and Salil Vadhan. Efficiency improvements in constructing pseudorandom generators from one-way functions. In *Proceedings of the 42nd ACM symposium on Theory of computing*, STOC '10, pages 437–446, New York, NY, USA, 2010. ACM.

[IP07] Yuval Ishai and Anat Paskin. Evaluating branching programs on encrypted data. In *Theory of Cryptography*, pages 575–594. Springer, 2007.

[IR89] R. Impagliazzo and S. Rudich. Limits on the provable consequences of one-way permutations. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, STOC '89, pages 44–61, New York, NY, USA, 1989. ACM.

[Kha93] Michael Kharitonov. Cryptographic hardness of distribution-specific learning. In *Proceedings of the twenty-fifth annual ACM symposium on Theory of computing*, STOC '93, pages 372–381, New York, NY, USA, 1993. ACM.

[LMN93] N. Linial, Y. Mansour, and N. Nisan. Constant-depth circuits, fourier transform and learnability. In *Journal of the ACM*, volume 40, 1993.

[LP11] Richard Lindner and Chris Peikert. Better key sizes (and attacks) for lwe-based encryption. In *Proceedings of the 11th international conference on Topics in cryptology: CT-RSA 2011*, CT-RSA'11, pages 319–339, Berlin, Heidelberg, 2011. Springer-Verlag.

[McE78] Robert J. McEliece. A Public-Key Cryptosystem Based on Algebraic Coding Theory. In *JPL Deep Space Network Progress Report*, volume 42–44, pages 114–116, 1978.

[MM11] Daniele Micciancio and Petros Mol. Pseudorandom knapsacks and the sample complexity of lwe search-to-decision reductions. In *Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference*, volume 6841 of *Lecture Notes in Computer Science*, page 461. Springer, 2011.

[MV11] Eric Miles and Emanuele Viola. On the complexity of non-adaptively increasing the stretch of pseudorandom generators. In *Proceedings of the 8th conference on Theory of cryptography*, TCC'11, pages 522–539, Berlin, Heidelberg, 2011. Springer-Verlag.

[MX10] M. Mahmoody and D. Xiao. On the power of randomized reductions and the checkability of sat. In *Proceeedings of 25th IEEE Conference on Computational Complexity (CCC)*, pages 64–75, 2010.

[Pai99] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Advances in Cryptology – Eurocrypt '99*, pages 223–238, 1999.

[Pei09] Chris Peikert. Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In *Proceedings of the 41th ACM Symposium on Theory of Computing*, pages 333–342, New York, NY, USA, 2009. ACM.

[Pin64] M. S. Pinsker. *Information and information stability of random variables and processes*. Holden-Day, 1964.

[RAD78] R. Rivest, L. Adleman, and M. Dertouzos. On data banks and privacy homomorphisms. In *Foundations of Secure Computation*, pages 169–177. Academic Press, 1978.

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, STOC '05, pages 84–93, New York, NY, USA, 2005. ACM.

[Reg09] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6), 2009.

[Rot11] Ron Rothblum. Homomorphic encryption: From private-key to public-key. In *Proceedings of the Theory of Cryptography Conference (TCC)*, pages 219–234, 2011.

[RSA78] Ronald L Rivest, Adi Shamir, and Len Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[RTV04] Omer Reingold, Luca Trevisan, and Salil Vadhan. Notions of reducibility between cryptographic primitives. In Moni Naor, editor, *Theory of Cryptography*, volume 2951 of *Lecture Notes in Computer Science*, pages 1–20. Springer Berlin Heidelberg, 2004.

[Sim82] Hans-Ulrich Simon. A tight $\log \log n$-bound on the time for parallel ram's to compute nondegenerated boolean functions. *Information and Control*, 55(1):102 – 107, 1982.

[Sim98] Daniel Simon. Finding collisions on a one-way street: Can secure hash functions be based on general assumptions? In Kaisa Nyberg, editor, *Advances in Cryptology EUROCRYPT'98*, volume 1403 of *Lecture Notes in Computer Science*, pages 334–345. Springer Berlin / Heidelberg, 1998. 10.1007/BFb0054137.

[SS92] V. M. Sidelnikov and S. O. Shestakov. On insecurity of cryptosystems based on generalized reed-solomon codes. *Discrete Mathematics and Applications*, 2(4):439–444, 1992.

[SV03] A. Sahai and S. Vadhan. A complete problem for statistical zero knowledge. *J. ACM*, 50:196–249, 2003.

[Val84] Leslie G. Valiant. Short Monotone Formulae for the Majority Function. *J. Algorithms*, 5(3):363–366, 1984.

[vDGHV10] M. van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan. Fully Homomorphic Encryption from Integers. In *Eurocrypt*, 2010.

[vN56] J. von Neumann. Probabilistic logics and synthesis of reliable organisms from unreliable components. In C. Shannon and J. McCarthy, editors, *Automata Studies*, pages 43–98, 1956.

[Wie10] Christian Wieschebrink. Cryptanalysis of the Niederreiter Public Key Scheme Based on GRS Subcodes. In *PQCrypto*, 2010.