UNIVERSITY OF CALIFORNIA SAN DIEGO

The Design of Efficient and Secure Lattice-based (FH)E

A dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy

in

Computer Science

by

Mark Douglas Schultz-Wu

Committee in charge:

Professor Daniele Micciancio, Chair
Professor Mihir Bellare
Professor Nadia Heninger
Professor Farinaz Koushanfar
Professor Tajana Rosing

2024

The Dissertation of Mark Douglas Schultz-Wu is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2024

DEDICATION

This work is dedicated to my wife, Renee Schultz-Wu.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

Thank you to my advisor, Daniele Micciancio, for his endless guidance and patience.

Thank you to my collaborators, Baiyu Li, Jessica Sorrell, and Mariana Raykova, for your insight and knowledge.

Thank you to my friends and colleagues at U.C.S.D., Vivek Arte, Michael Borkowski, Marco Carmosino, Hannah Davis, Gabrielle De Micheli, Nick Genise, Felix Günther, Miro Haller, Max Hopkins, Kaave Hosseini, Joseph Jaeger, Amy Kanne, Rex Lei, Sihan Liu, Dylan Lukes, Gaurav Mahajan, Ruth Ng, Anthony Ostuni, Rishabh Ranjan, Sankeerth Rao, Doreen Riepel, Keegan Ryan, Laura Shea, Mary Anne Smart, George Sullivan, Jiahao Sun, Sophia Sun, Udayan Tandon, Marysia Tran, Psi Vesely, Christopher Ye, and Jiapeng Zhang.

Thank you to my friends outside of U.C.S.D., K. Makana Castillo-Martin, Ira Globus-Harris, Andie Hoshijo, Julia Len, Jade Mabee, and Michael Rosenberg.

I would also like to thank my parents, Margot and Stan, my siblings (and sibling-in-law), Katie, Sean, and Sean, and especially my wife Renee.

Chapter 2, in full, is a reprint of the material as it appears in Theory of Cryptography 2024. Micciancio, Daniele; Schultz-Wu, Mark. "Bit Security: Optimal adversaries, Equivalence results, and a Toolbox for Computational/Statistical Security Analysis". The dissertation author was a primary investigator and author of this material.

Chapter 3, in full, is a reprint of the material as it appears in Advances in Cryptology — CRYPTO 2023. Micciancio, Daniele; Schultz-Wu, Mark. "Error Correction and Ciphertext Quantization in Lattice Cryptography". The dissertation author was a primary investigator and author of this material.

Chapter 4, in full, is a reprint of the material as it appears in Advances in Cryptology — CRYPTO 2022. Li, Baiyu; Micciancio, Daniele; Schultz-Wu, Mark; Sorrell, Jessica. "Securing Approximate Homomorphic Encryption Using Differential Privacy". The dissertation author was a primary investigator and author of this paper.

VITA

2018 Bachelor of Arts, Mathematics/Computer Science, Reed College

2021 Masters of Science, Computer Science, University of California San Diego

2024 Doctor of Philosophy, Computer Science, University of California San Diego

ABSTRACT OF THE DISSERTATION

The Design of Efficient and Secure Lattice-based (FH)E

by

Mark Douglas Schultz-Wu

Doctor of Philosophy in Computer Science

University of California San Diego, 2024

Professor Daniele Micciancio, Chair

Lattice-based cryptography leverages Euclidean lattices (and carefully-applied "noise") to construct secure cryptographic primitives. In recent years, these primitives have become quite practical (and academically popular), yielding a large number of variant schemes that are mild variants on the same core construction(s).

First, we introduce a relaxed notion of security for a cryptographic primitive that we call (c,s)-*bit security*. This parameterizes security with a (standard, *computational*) security parameter c, as well as a *statistical* security parameter s, and seems well-adapted for summarizing the concrete hardness of problems that contain both computationally-hard and statistically-hard components. We pair this with the notion of distinguishing advantage of aborting adversaries (Micciancio and Walter, Eurocrypt 2018), and characterize optimal adversaries in this setting.

Next, we propose a framework for the design of lattice-based encryption, parameterized by two coding-theoretic objects. We show that one can instantiate many lattice-based cryptosystems with compact ciphertexts in our framework, and show there are fundamental limits on the ciphertext size for cryptosystems built within our framework.

Finally, we show that one may harden the approximate FHE scheme of Cheon, Kim, Kim, and Song (Asiacrypt 2017) against the passive attacks of Li and Micciancio (Eurocrypt 2021), via applying an appropriate notion of differential privacy. Here, we find that to achieve (c, s)-bit security, the overhead of our countermeasure scales entirely with s (which may plausibly be set lower than c). We show that our countermeasure's overhead is nearly optimal, by arguing that instantiating it with smaller overhead yields an insecure scheme. Finally, we investigate another proposed countermeasure that lacked a proof of security, and show simple attacks against it.

Chapter 1

Preliminaries

For $n \in \mathbb{N}$, we will frequently use the notation $[n] := \{0, 1, \ldots, n-1\}$. For a finite set S, we write $|S|$ for the cardinality of S. For a continuous subset $S \subseteq \mathbb{R}^n$, we write $\mathsf{vol}(S)$ for the volume (Lebesgue measure) of S. Our continuous subsets will always be "nice" (compact and convex), such that their volume is well-defined. We write $f(S) = \{f(x) : x \in S\}$ for the image of a set $S \subseteq A$ under a function $f : A \to B$, and $X + Y = \{x + y \mid x \in X, y \in Y\}$ for the (Minkowski) sum of two subsets $X, Y \subseteq A$ of an abelian group $(A, +, 0)$. We will write $r \cdot \mathscr{B}_n$ for the Euclidean ball of radius r, centered at 0, and $r \cdot \mathscr{B}_n^{(\infty)} = [-r, r]^n$ for the ℓ_∞ ball of radius r. We will write $r \cdot \mathscr{B}_n^{(p)}$ to uniformly refer to either of these objects (but omit p for the more common Euclidean case).

1.1 Linear Algebra

Throughout, we will write scalars a as lower-case italicized, vectors $\mathbf{a}$ as lower-case bolded, matrices $\mathbf{A}$ as upper-case bolded. We write $[\mathbf{A}, \mathbf{B}]$ for horizontal concatenation of matrices, and $(\mathbf{A}, \mathbf{B}) = [\mathbf{A}^t, \mathbf{B}^t]^t$ for vertical concatenation.

1.2 Algebraic Number Theory

For any positive integer N, let $\Phi_N(X) = \prod_{j \in \mathbb{Z}_N^*} (X - \omega_N^j)$ be the Nth cyclotomic polynomial, where $\omega_N = e^{2\pi \mathsf{i}/N} \in \mathbb{C}$ is the complex Nth principal root of unity, and $\mathbb{Z}_N^*$ is the group of invertible integers modulo N. We recall that $\Phi_N(X) \in \mathbb{Z}[X]$ is a monic polynomial of degree $n = \varphi(N) = |\mathbb{Z}_N^*|$ with integer coefficients. We denote by $R^N = \mathbb{Z}[X]/(\Phi_N(X))$ the ring of integers of the

number field $\mathbb{Q}[X]/(\Phi_N(X))$, omitting the superscript when it is clear from context. We use $R_Q^N = \mathbb{Z}[X]/(Q, \Phi_N(X))$ to denote the ring of elements of R^N reduced modulo Q. We write an element of these rings as $\mathscr{a}$, e.g. as lower-case and curly. Typically[1], these will be elements of the ring R_Q^N, rather than R^N.

An element $\mathscr{a} \in \mathbb{R}[X]/(\Phi_N(X))$ may be embedded into $\mathbb{C}^n$ under the *canonical embedding* $\tau(\mathscr{a})$ (typically defined over $\mathbb{Q}[X]/(\Phi_N(X))$, but naturally extending to $\mathbb{R}[X]/(\Phi_N(X))$). The map $\tau(\mathscr{a})$ takes $\mathscr{a}$ to the $n = \varphi(N)$ evaluations of a at the n roots of $\Phi_N(X)$. We may occasionally write $\widehat{a} := \tau(\mathscr{a})$ as shorthand for these values. Notice that these n values come in conjugate pairs and can be identified as a vector in $\mathbb{C}^{n/2}$ via a projection $\pi : (z, \bar{z}) \mapsto z$.

1.3 Probability Theory

For a random variable X and point y, we write $X(y)$ as shorthand for the probability that y occurs, e.g. the probability mass function of a discrete random variable, or probability density function of a continuous random variable. For a set S, we write $X[S]$ for $\sum_{y \in S} X[y]$ (if X is discrete) or $\int_y X[y]\mathrm{d}y$ (if X is continuous). In doing so, we assume our random variables X are sufficiently "nice", which may be informally[2] summarized as

- either purely discrete, or purely continuous, and
- if purely continuous, sufficiently nice such that a probability density exists.

Throughout, we will often identify a random variable $X \leftarrow \mathscr{D}$ samples from some distribution $\mathscr{D}$, and the distribution itself, e.g. we may write $\mathscr{D}[S]$ for the probability that $X \in S$, where $X \leftarrow \mathscr{D}$. If S is a finite set, then $X \leftarrow S$ chooses X at random from S with uniform distribution.

We will work with the following distributions. Throughout, let $\rho(x) = \exp(-x^2/2)$ be the Gaussian kernel, normalized using the convention common in probability theory[3].

[1]The main exception to this is during "error analysis" in lattice-based cryptography. While we investigate the impact of errors on the sizes of lattice-based ciphertexts in detail in Chapter 3, this is for (standard) LWE-based encryption, not (ring) LWE-based encryption. We perform some ring-based error analysis in Chapter 4 (see for example Section 4.4.2), though this occupies a relatively minor part of the work included in this dissertation.

[2]Formalizing this is not particularly hard, but also distracts from the cryptographic focus of our work.

[3]There is a competing normalization $\exp(-\pi x^2)$ that is often used in Physics, as it behaves better with respect to

Definition 1 (Bernoulli Distribution)**.** *Let $p \in [0,1]$. The* Bernoulli *distribution of parameter p (notated $\mathsf{Bern}(p)$) is the discrete distribution on $\{0,1\}$ with probability mass function $\Pr_{X \leftarrow \mathsf{Bern}(p)}[X = 0] = p$, $\Pr_{X \leftarrow \mathsf{Bern}(p)}[X = 1] = 1 - p$.*

We will occasionally refer to the Discrete Gaussian distribution. We refer the interested reader to [14, 31] (and their references) for more details on this distribution.

Definition 2 (Discrete Gaussian Distribution)**.** *Let $n \in \mathbb{N}$, and let $\sigma > 0$. The (mean-zero)* Discrete Gaussian *of parameter σ^2 (notated $\mathscr{N}_{\mathbb{Z}^n}(0, \sigma^2 I_n)$) is the discrete probability distribution supported on $\mathbb{Z}^n$ with probability mass function proportional to $\rho(\|\mathbf{x}\|_2^2/\sigma^2)$.*

Definition 3 (Discrete Uniform Distribution)**.** *Let S be a finite set. The Discrete Uniform distribution on S (notated $\mathsf{Unif}(S)$) is the discrete probability distribution supported on S with probability mass function proportional to $1/|S|$.*

We will leverage several continuous distributions as well.

Definition 4 (Continuous Gaussian Distribution)**.** *Let $n \in \mathbb{N}$, and let $\sigma > 0$. The (mean-zero)* Continuous Gaussian *of parameter σ^2 (notated $\mathscr{N}(0, \sigma^2 I_n)$) is the continuous probability distribution supported on $\mathbb{R}^n$ with probability density function proportional to $\rho(\|\mathbf{x}\|_2^2/\sigma^2)$.*

Definition 5 (Continuous Uniform Distribution)**.** *Let $S \subseteq \mathbb{R}^n$ be a compact, convex set. The* Continuous Uniform *distribution on S (notated $\mathsf{Unif}(S)$) is the continuous probability distribution supported on S with probability density function proportional to $1/\mathsf{vol}(S)$.*

We will associate with any random vector $\mathbf{x}$ a mean $\mu_{\mathbf{x}} := \mathbb{E}[\mathbf{x}]$ and Covariance matrix $\Sigma_{\mathbf{x}} := \mathsf{Cov}(\mathbf{x})_{i,j} := \mathbb{E}[(\mathbf{x} - \mu_{\mathbf{x}})_i (\mathbf{x} - \mu_{\mathbf{x}})_j]$. Both quantities behave well with respect to linear transformations.

Lemma 1. *Let $\mathbf{x} \leftarrow \mathscr{D}$ be a random vector on $\mathbb{R}^n$. Let $\mathbf{A} \in \mathbb{R}^{m \times n}$, and let $\mathbf{y} \in \mathbb{R}^m$ Then $\mathbf{A}\mathbf{x} + \mathbf{y}$ is a random vector with mean $\mathbf{A}\mu_{\mathbf{x}} + \mathbf{y}$, and covariance $\Sigma_{\mathbf{y}} = \mathbf{A}\Sigma_{\mathbf{x}} A^t$.*

Fourier transforms. It often occurs in lattice cryptography as well, see for example [31].

We highlight the class of random vectors with mean 0 and covariance I_n, which we call *isotropic*. Any random variable $\mathbf{x} \leftarrow \mathscr{D}$ may be mapped by an affine transformation to an isotropic random variable, completely analogously[4] to how any (potentially multivariate) continuous Gaussian may be mapped to an i.i.d. collection of univariate standard (meaning mean zero and variance 1) Gaussians.

1.3.1 Similarity Measures between Distributions

We use several similarity measures between probabilistic distributions. Below, we include formula for nearly[5] all similarity measures we use in the setting of discrete random variables, though the extension to continuous random variables is standard, see [73, Chapter 7].

Definition 6. *Let $\mathscr{D}_0, \mathscr{D}_1$ be distributions on a set Ω. Define*

- *Statistical Distance:* $\Delta_{\mathsf{SD}}(\mathscr{D}_0, \mathscr{D}_1) = \frac{1}{2}\sum_{x\in\Omega} |\mathscr{D}_0(x) - \mathscr{D}_1(x)|$,
- *(Squared) Hellinger Distance:* $\Delta^2_{\mathsf{H}}(\mathscr{D}_0, \mathscr{D}_1) = \frac{1}{2}\sum_{x\in\Omega}(\sqrt{\mathscr{D}_0(x)} - \sqrt{\mathscr{D}_1(x)})^2$
- *Kullback-Liebler Divergence:* $D(\mathscr{D}_0||\mathscr{D}_1) := \sum_{x\in\Omega} \mathscr{D}_0(x) \ln\left(\frac{\mathscr{D}_0(x)}{\mathscr{D}_1(x)}\right)$
- *Renyi Divergence of Order* $1/2$*:* $\Delta_{\mathsf{R};1/2}(\mathscr{D}_0||\mathscr{D}_1) = -2\ln\sum_{x\in\Omega}\sqrt{\mathscr{D}_0(x)\mathscr{D}_1(x)} = -2\ln(1 - \Delta^2_{\mathsf{H}}(\mathscr{D}_0, \mathscr{D}_1))$,
- *(Squared) Le Cam Distance:* $\Delta^2_{\mathsf{LC}}(\mathscr{D}_0, \mathscr{D}_1) = \frac{1}{2}\sum_{x\in\Omega}\frac{(\mathscr{D}_0(x)-\mathscr{D}_1(x))^2}{\mathscr{D}_0(x)+\mathscr{D}_1(x)}$.

Everything we call a "distance" above is a true distance (e.g. a metric). We next summarize several other properties that all of the above divergences satisfy.

Lemma 2. *Let $\delta \in \{\Delta_{\mathsf{SD}}, \Delta^2_{\mathsf{H}}, D, \Delta_{\mathsf{R};1/2}, \Delta^2_{\mathsf{LC}}\}$ be a divergence of Definition 6. Let $\mathscr{D}_0, \mathscr{D}_1$ be any distributions on a set Ω. Then*

1. $\delta(\mathscr{D}_0||\mathscr{D}_1) \geq 0$,

[4]Note that in the non-Gaussian case, the isotropic random variable $\mathbf{x}$ only has *uncorrelated* coordinates. G

[5]The exception is the the "Bit Security Divergence", introduced in Eq. (2.4), as the definition of this similarity measure was a novel contribution of Chapter 2.

2. $\delta(\mathscr{D}_i||\mathscr{D}_i) = 0$,

3. $(\mathscr{D}_0, \mathscr{D}_1) \mapsto \delta(\mathscr{D}_0||\mathscr{D}_1)$ *is a jointly convex function, and is therefore additionally convex in each input separately,*

4. *Data Processing Inequality: for any randomized transformation A,* $\delta(A(\mathscr{D}_0)||A(\mathscr{D}_1)) \leq \delta(\mathscr{D}_0||\mathscr{D}_1)$

Proof. These all immediately follow from viewing the divergences of Definition 6 as what are known as f-divergences[6]. We direct the interested reader to [73, Section 7]. □

The above similarity measures are generally related as follows.

Lemma 3 ([73, Section 7].)**.** *For any two distributions* $\mathscr{D}_0, \mathscr{D}_1$ *we have*

$$\Delta^2_{\mathsf{H}}(\mathscr{D}_0, \mathscr{D}_1) \leq \Delta_{\mathsf{SD}}(\mathscr{D}_0, \mathscr{D}_1) \leq \sqrt{2}\Delta_{\mathsf{H}}(\mathscr{D}_0, \mathscr{D}_1)$$
$$\Delta_{\mathsf{H}}(\mathscr{D}_0, \mathscr{D}_1) \leq \Delta_{\mathsf{LC}}(\mathscr{D}_0, \mathscr{D}_1) \leq \sqrt{2}\Delta_{\mathsf{H}}(\mathscr{D}_0, \mathscr{D}_1)$$
$$\Delta^2_{\mathsf{SD}}(\mathscr{D}_0, \mathscr{D}_1) \leq 2\Delta^2_{\mathsf{H}}(\mathscr{D}_0, \mathscr{D}_1) \leq D(\mathscr{D}_0||\mathscr{D}_1).$$

In other words, Δ^2_{H} and Δ^2_{LC} are (up to constant factors, e.g. tightly) equivalent, but Δ_{SD} and (either of Δ^2_{H} or Δ^2_{LC}) are only loosely equivalent. $\Delta^2_{\mathsf{H}}, \Delta^2_{\mathsf{LC}}, \Delta_{\mathsf{SD}}$ are all at most 1. $\Delta_{\mathsf{R};1/2}$ is equivalent to Δ^2_{H} and Δ^2_{LC}, under certain technical conditions which need not always hold[7]. D is generally[8] not equivalent to other distance measures.

$\Delta_{\mathsf{R};1/2}$ is a monotone transformation of Δ^2_{H}, but may as well be bounded by Δ^2_{H} as follows.

Lemma 4. *For any two distributions* $\mathscr{D}_0, \mathscr{D}_1$ *such that* $\Delta_{\mathsf{R};1/2}(\mathscr{D}_0, \mathscr{D}_1) < \infty$*, we have*

$$\Delta^2_{\mathsf{H}}(\mathscr{D}_0, \mathscr{D}_1) \leq \frac{1}{2}\Delta_{\mathsf{R};1/2}(\mathscr{D}_0, \mathscr{D}_1) \leq \frac{\Delta^2_{\mathsf{H}}(\mathscr{D}_0, \mathscr{D}_1)}{1 - \Delta^2_{\mathsf{H}}(\mathscr{D}_0, \mathscr{D}_1)} \leq 2\Delta^2_{\mathsf{H}}(\mathscr{D}_0, \mathscr{D}_1),$$

[6] f-divergences may be characterized as the class of *decomposable* divergences (meaning $\delta(\mathscr{D}_0, \mathscr{D}_1) = \sum_x f(\mathscr{D}_0(x), \mathscr{D}_1(x))$ for some f) such that data-processing inequality holds, see [42].

[7] For example, if the supports of $\mathscr{D}_0$ and $\mathscr{D}_1$ are disjoint, then $\Delta_{\mathsf{R};1/2}(\mathscr{D}_0||\mathscr{D}_1) = \infty$. This additionally holds "asymptotically". In particular, if $\mathscr{D}_0^\varepsilon := (1-\varepsilon, 0, \varepsilon)$, and $\mathscr{D}_1^\varepsilon := (0, 1-\varepsilon, \varepsilon)$, one can compute that $\Delta_{\mathsf{R};1/2}(\mathscr{D}_0^\varepsilon||\mathscr{D}_1^\varepsilon) = \ln(1/\varepsilon^2)$ approaches ∞ as $\varepsilon \to 0$.

[8] In very limited settings one may establish equivalence, generally under the name of a "Reverse Pinsker Inequality". See [79] for details.

where the last inequality assumes $\Delta^2_{\mathsf{H}}(\mathscr{D}_0, \mathscr{D}_1) \leq 1/2$.

Proof. Easily follows from the bounds $1 - (1/t) \leq \ln t \leq t - 1$ and relation $\Delta_{\mathsf{R};1/2}(\mathscr{D}_0, \mathscr{D}_1) = -2\ln(1 - \Delta^2_{\mathsf{H}}(\mathscr{D}_0, \mathscr{D}_1))$. See [85] for details. □

All f-divergences have what are known as *variational representations*, or ways of writing them as the suprema of a certain functional over an explicit class of functions. A general presentation of this theory requires background in convex geometry that we avoid for simplicity — see [73, Chapter 7] for details. We will solely use the following variational representations[9].

Lemma 5. *Let* $\mathscr{D}_0, \mathscr{D}_1$ *be discrete probability distributions on some set* Ω*. Then*

- $\Delta_{\mathsf{SD}}(\mathscr{D}_0, \mathscr{D}_1) = \sup_{g:|g|\leq 1/2} \mathbb{E}_{\mathscr{D}_0}[g(x)] - \mathbb{E}_{\mathscr{D}_1}[g(x)]$,
- $\Delta^2_{\mathsf{LC}}(\mathscr{D}_0, \mathscr{D}_1) = \sup_{g:\Omega\to\mathbb{R}} \frac{(\mathbb{E}_{\mathscr{D}_0}[g(x)] - \mathbb{E}_{\mathscr{D}_1}[g(x)])^2}{\mathsf{Var}_{\mathscr{D}_0}(g(x)) + \mathsf{Var}_{\mathscr{D}_1}[g(x)]}$.

Note that in both cases a maximizer may be explicitly described, namely for Δ_{SD} $g(x)$ is (half) the indicator function of $\{x \mid \mathscr{D}_0(x) > \mathscr{D}_1(x)\}$, while for Δ^2_{LC}, one has $g(x) = \frac{\mathscr{D}_0(x) - \mathscr{D}_1(x)}{\mathscr{D}_0(x) + \mathscr{D}_1(x)}$.

A probability ensemble $\{\mathscr{D}^\theta\}_\theta$ is a family of distributions indexed by an arbitrary parameter θ. Given a similarity measure δ between distributions, we extend it to distribution ensembles via $\delta(\{\mathscr{D}^\theta_0\}_\theta, \{\mathscr{D}^\theta_1\}_\theta) = \sup_\theta \delta(\mathscr{D}^\theta_0, \mathscr{D}^\theta_1)$. We will write $\mathscr{D}^{\otimes q} = (\mathscr{D}, \mathscr{D}, \ldots, \mathscr{D})$ as q independent and identically distributed (i.i.d.) copies of a distribution $\mathscr{D}$. We abbreviate a list of random variables $(\mathscr{D}_1, \ldots, \mathscr{D}_n)$ as $(\mathscr{D}_i)_i$. For such a list, we write $\mathscr{D}_{<i}$ to denote the prefix $(\mathscr{D}_1, \ldots, \mathscr{D}_{i-1})$.

Lemma 6 (Chain Rule for the KL Divergence, Theorem 2.2 of [72])**.** *The KL divergence satisfies If* $(\mathscr{D}_0, \mathscr{D}_1)$ *and* $(\mathscr{D}'_0, \mathscr{D}'_1)$ *are pairs of (possibly dependent) random variables, then*

$$\begin{aligned} D((\mathscr{D}_0, \mathscr{D}_1)||(\mathscr{D}'_0, \mathscr{D}'_1)) &\leq \mathbb{E}_{x\sim\mathscr{D}_0}[D((\mathscr{D}_1 \mid \mathscr{D}_0 = x)||(\mathscr{D}'_1 \mid \mathscr{D}'_0 = x))] + D(\mathscr{D}_0||\mathscr{D}'_0) \\ &\leq \max_{x\in\mathsf{supp}(\mathscr{D}_0)} ((\mathscr{D}_1 \mid \mathscr{D}_0 = x)||(\mathscr{D}'_1 \mid \mathscr{D}'_0 = x)) + D(\mathscr{D}_0||\mathscr{D}'_0). \end{aligned}$$

[9] Note that the variational representation of Δ^2_{LC} is not directly found in [73]. It can be obtained by writing Δ^2_{LC} in terms of another distance measure (the χ^2 divergence), for which [73] does list a variational representation.

Note that there exists a variant of the above for other divergences, though this is most easily described using the theory of Renyi divergences[10] that this work will not require. See [73, Eq. (7.77)].

We introduce the following notation to more compactly bound the divergence between vectors of random variables.

Definition 7. *Let $\mathscr{D} = (\mathscr{D}_i)_{i=1}^n, \mathscr{D}' = (\mathscr{D}'_i)_{i=1}^n$ be two lists of discrete random variables over the support $\prod_{i=1}^n \Omega_i \subseteq \mathbb{R}^n$, and δ any divergence. We define the* vector divergence $\hat{\delta}(\mathscr{X}||\mathscr{Y})$ *to be the non-negative real vector $(v_1, \ldots, v_n) \in \mathbb{R}^n_{\geq 0}$ with coordinates $v_i = \max \delta([\mathscr{D}_i \mid \mathscr{D}_{<i} = a]||[\mathscr{D}'_i \mid \mathscr{D}'_{<i} = a])$.*

In this notation, chain rule of the KL divergence (for example) can be written as $D(\mathscr{X}||\mathscr{Y}) \leq \|\widehat{D}(\mathscr{X}||\mathscr{Y})\|_1$.

1.4 Cryptography

1.4.1 The Learning with Errors Problem

Much of lattice cryptography relies on the hardness of the *learning with errors* problem.

Definition 8 (LWE problem)**.** *Let $m = n^{O(1)}$, and let $q \in [n^{O(1)}, 2^{O(n)}]$. Let χ_{sk} be a distribution on $\mathbb{Z}_q$, and χ_e be a distribution on $\mathbb{R}_q$. The Learning with Errors problem $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}$ is to distinguish the distribution $(\mathbf{A}, \mathbf{As}+\mathbf{e})$ from $(\mathbf{A}, \mathbf{U})$, where $\mathbf{A} \leftarrow \mathbb{Z}_q^{m\times n}$, $\mathbf{s} \leftarrow \chi_{\mathsf{sk}}^n$, and $\mathbf{e} \leftarrow \chi_e^m$, and $\mathbf{u} \leftarrow \mathbb{R}_q^m$.*

We rely on LWE where $\mathbf{e} \leftarrow \chi_e$ and $\mathbf{u} \leftarrow \mathbb{R}_q^m$ are *real* random variables (modulo q) to simplify our analysis. We omit the inclusion of m in the notation $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}$, as it has minimal impact on the hardness of the problem. The primary justification for the hardness of $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}$ is that it admits reductions from worst-case hard lattice problems, initially due to Regev [76].

Theorem 1. *For any $m = n^{O(1)}$, any modulus $q \leq 2^{n^{O(1)}}$, let χ_e be any (discretized) Gaussian distribution χ of parameter $\sigma \geq 2\sqrt{n}$, and χ_{sk} be the uniform distribution on $\mathbb{Z}_q$. Then solving*

[10] Many of our divergences *are* closely related to Renyi divergences, namely Δ^2_{H}, Δ^2_{LC}, D, and $\Delta_{\mathsf{R};1/2}$. Still, we find that it will always suffice to first bound our divergences by the KL divergence, and appeal to the chain rule for this divergence.

the decision $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}$ *problem is at least as hard as quantumly solving* GapSVP_γ *and* SIVP_γ *on arbitrary n-dimensional lattices, where* $\gamma = \tilde{O}(nq/\sigma)$.

This work will not need definitions of of GapSVP_γ and SIVP_γ. We call attention to the approximation factor $\gamma = \tilde{O}(nq/\sigma)$, which controls the hardness of the problem, and depends on the "modulus to noise" ratio q/σ. The Gaussian parameter can often be set to a fixed polynomial $\sigma = 2\sqrt{n}$, so that larger values of q result in constructions that are both less efficient and less secure. Of particular interest will be the cases of *polynomial* $q/\sigma = n^{O(1)}$, and *superpolynomial* $q/\sigma = n^{\omega(1)}$ modulus to noise ratio.

Chapter 2

On Mixing Computational and Statistical Bit Security

2.1 Chapter Introduction

The level of security provided by a cryptographic construction is customarily measured in "bits". The intuition is that breaking an application offering "n bits of security" should have a cost[1] comparable to mounting a key recovery attack on an ideal cryptographic function with a key space of size 2^n. Formalizing this intuition is not entirely trivial, because cryptographic attacks often exhibit a trade-off between the cost (e.g., the running time T_A) of the attack, and its success probability ε_A. For (verifiable) search problems, like forging digital signatures, it is well established[2] that bit security can be defined as the quantity $\log_2(T_A/\varepsilon_A)$, minimized over all possible adversaries A. However, the situation for decision problems (like indistinguishability of ciphertexts, zero knowledge, pseudorandomness, etc). is far less clear. We recall that in a decision game the goal of the adversary is to distinguish between two distributions $\mathscr{D}_b$ for $b \in \{0,1\}$. So, a naive approach to measure security could be to mimic the definition for search problems, and replace the quantity $\log_2(T_A/\varepsilon_A)$ with $\log_2(T_A/\delta_A)$, where $\delta_A = 2\varepsilon_A - 1$ is the advantage (over a random choice) of guessing the bit b. But it is well known that this naive definition leads to paradoxical situations, where for example [26] an algorithm G is deemed more secure (i.e., it is attributed a

[1] Various measures of cost have been considered, and the reader is referred to [8, Appendix B] for a discussion. For simplicity, in this paper we identify the cost of an attack with its running time.

[2] This is justified by the fact that one can repeat the attack $O(1/\varepsilon)$ times to make the success probability arbitrarily close to 1.

higher level of bit security) as a pseudorandom generator than as a one-way function. This is at odds with cryptographic intuition because pseudorandomness is a stronger security requrement than one-wayness. (See [61] and references therein for a detailed discussion of this and other problematic examples).

During the last few years, several papers have investigated the problem of giving meaningful definitions of bit security [61, 84, 85, 53, 49], or using them to give a tight security analysis of cryptographic primitives (e.g., see [1, 53]). The first work to propose a satisfactory definition of bit security for decision games is the one of Micciancio and Walter [61]. A key element of their definition is to consider attackers that may output either a bit $b \in \{0,1\}$ (indicating a decision between $\mathscr{D}_0$ and $\mathscr{D}_1$) or a special "don't know" symbol $\perp$. Interestingly, [61] shows that this simple extension of traditional adversaries, together with an appropriate definition of advantage (already used by [51] in a different context,) allows to resolve all the previously mentioned paradoxes, and argue (by means of examples) that this is the right definition of bit security.[3] Since then, a number of alternative definitions have appeared [84, 85, 53, 49], with various motivations. Watanabe and Yasunaga [84] proposed an alternative framework to define bit security that directly admits what they call an "operational interpretation", and later argued [85] that it is equivalent to the original MW definition [61]. A seemingly attractive feature of their definition is that it only requires standard (non-aborting) adversaries with output in $\{0,1\}$. A variant of their definition that (similarly to [61]) interpolates between search and decision problems is given in [49]. In a different and orthogonal direction, Li at al. [53] extend the MW definition to encompass both computational and statistical security. Informally, statistical security provides a strong measure of the security even against computationally unbounded adversaries. When achievable, statistical security has the advantage of being easier to evaluate, and not requiring any computational assumptions. In practice, when setting parameters and optimizing efficiency, it is common to require lower levels of statistical bit security s, than computational bit security c. For example, $s = 80$ is usually considered more

[3] This is at least for decision problems. The work [61] also proposes a more general definition based on information theory that interpolates between search and decision problems, but the corresponding notion of bit security for intermediate cases is largely unexplored. In this paper, we focus on the special case of decision problems which is the most relevant to cryptography.

than acceptable, while computational security typically requires $c \geq 128$ or even higher values to anticipate possible improvements in the computational complexity of attacks. Li at al. [53] define (c,s)-security as satisfied by a protocol that provides *either* c bits of computational security, *or* s bits of statistical security against any possible attack. We remark that a protocol can admit both attacks with running time much less than 2^c (as long as their advantage is less than 2^{-s}) and (different) attacks achieving advantage very close to 1 (as long as their running time is higher than 2^c). In other words, a (c,s)-secure protocol can achieve neither c-bits of computational security nor s-bits of statistical security. Still, morally, it provides an acceptable level of security whenever s-bit statistical security and c-bit computational security are considered individually adequate. The advantage of (c,s)-security is that it allows to seamlessly combine statistical and computational cryptographic primitives (something very common in practice) and still be able to formally quantify the security level of an application. However, the notion of (c,s)-security has not been further explored, and, despite its potential usefulness, it has seen little adoption due to the lack of tools to simplify its usage.

Our Contributions and Techniques

In this work, we examine the bit security definitions of [61, 84, 53], proving structural results about optimal adversaries in the statistical setting, clarifying the relation between the MW and WY bit security definitions, and then applying these results to the recent notion of (c,s)-bit security. Our main contributions, described in more details in the next subsections, can be summarized as follows:

- We characterize the MW adversaries achieving the optimal bit-security advantage. Specifically, we show that these adversaries may be assumed to be deterministic (Corollary 1) and have a special "threshold" structure (Theorem 4).

- We show (Theorem 6) that the WY notion of bit security is equivalent to the original MW bit security definition. We remark that a proof of this equivalence was already given in [85], but, as we are going to describe, that proof contains a gap. We clarify the relation between the two definitions by filling the gap and also giving a simpler proof of the equivalence.

- Despite the fact that the WY definition only uses traditional (non-aborting) adversaries, we show (Theorem 7) that the natural "maximum likelihood" distinguisher can offshoot the correct bit security level by a large margin. So, the advantages of using standard (non-aborting) adversaries in the characterization of bit security put forward in [84] are unclear.

- We show that common proof techniques widely used in the analysis of cryptographic protocols can be extended to work with the more general notion of computational-statistical security from [53]. Specifically, we show that (c,s)-security fully supports the use of hybrid arguments (Theorem 8) and probability substitution (Theorem 9).

On the technical side, many of our results rely on a new class of adversaries that further extends the MW (aborting) adversaries, and that may be of independent interest. Specifically, we make use of adversaries (for decision games) that may output not just $0,1$ (represending a high confidence decision) or $\perp$ (represending no confidence), but an arbitrary value $\sigma \in [-1,1]$, with the sign $\sigma/|\sigma| \in \{-1,1\}$ representing the decision, and the magnitude $|\sigma| \in [0,1]$ the confidence level that can vary continuously from 0 (no confidence) to 1 (perfect confidence). Interestingly, we show (Theorem 2) that these "fuzzy" adversaries still define precisely the same notion of bit security as the standard MW "aborting" adversaries. Still, having the output vary continuously allows the use of analytical techniques, and it is useful to prove the results in this paper. We believe that the characterization of bit security in terms of these more general fuzzy adversaries may find other applications, and is of independent interest.

Related Work

As mentioned, our work directly builds on the bit security frameworks of [61, 84], so is directly related to these works. Our work is also tangentially related to the bit security framework of [49], though this work mostly focuses on generalizing (a variant of) the framework of [84] to non-decision games, whereas we focus on decision games. Our work on the optimal adversary for the MW advantage is additionally related to the notion of (binary) hypothesis testing with an aborting option, see for example [36], though the measure optimized in that work does not appear

to be related to the MW advantage. The similarity between our work and binary hypothesis testing with a rejection option is perhaps more obvious from [47, Section 4], where (in a slightly different setting) optimality of threshold distinguishers was also highlighted. Finally, our discussion of the implications of [69] to the WY formulation of bit security is obviously heavily related to [69].

Paper organization

The rest of this paper is organized as follows. In the rest of this section we give a more detailed, still informal, description of our results and techniques. Section 2.2 defines the notation and preliminary results used in this paper. In Section 2.3 we formally define fuzzy adversaries, establish their equivalence (in both the computational and statistical setting) with the aborting adversaries of [61], and then use them to investigate the structure the MW adversaries achieving the optimal advantage. In Section 2.4, we explore the WY bit security definition and its equivalence with the MW bit security. Finally, in Section 2.5, we build our toolbox for the use of (c,s)-security in the analysis of cryptographic protocols, establishing the validity of hybrid arguments and probability replacement theorems. Section 2.6 concludes with some final remarks and open problems.

2.1.1 The Micciancio-Walter Advantage

Consider the problem of distinguishing between two distributions $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_1)$ over a set Ω. (Everything applies more generally to the case of more complex decision games where an adversary interacts with one of two oracles). Micciancio and Walter (following [51]) define the advantage of an "aborting" adversary $A : \Omega \to \{0,1,\perp\}$ as

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) = \frac{(\beta_A - \bar{\beta}_A)^2}{\beta_A + \bar{\beta}_A}, \tag{2.1}$$

where $\beta_A = \Pr_[A(x_b) = b]$ and $\bar{\beta}_A = \Pr[A(x_b) = 1-b]$ are the probability that A outputs the correct or incorrect bit, respectively, when $b \in \{0,1\}$ is chosen at random and $x_b \leftarrow \mathscr{D}_b$. For traditional (non-aborting) adversaries with output in $\{0,1\}$, we have $\beta + \bar{\beta} = 1$, and it is well known (and quite intuitive) that the best advantage is achieved by an adversary $A(x)$ that on input a sample

$x \in \Omega$, outputs the bit $b := \arg\max_{b' \in \{0,1\}} \mathscr{D}_{b'}(x)$ for which $\mathscr{D}_b(x)$ is highest. Moreover, the resulting optimal advantage equals precisely the square $\Delta_{\mathsf{SD}}(\mathscr{D}_0, \mathscr{D}_1)^2$ of the statistical distance between the two distributions. This allows to easily compute the bit security of $\mathscr{X}$ whenever the probability distributions are efficiently computable. This is a common scenario in cryptography, where, for example $\mathscr{D}_0$ may be an ideal probability distribution used in the theoretical analysis of a cryptographic scheme (e.g., a discrete gaussian distribution in lattice-based cryptography) and $\mathscr{D}_1$ is an approximate (more efficiently samplable) version of $\mathscr{D}_0$ used when implementing the algorithm in practice (e.g. using floating point numbers). In fact, this was precisely the motivation in [59, 61].

However, once the adversary is allowed to output $\perp$, it is no longer clear how to determine an optimal adversarial strategy, even when the probability distributions $\mathscr{D}_0, \mathscr{D}_1$ are efficiently computable. One of our main goals is to characterize the optimal aborting adversarial strategies, both to improve our understanding of the MW bit security definition, and offer a simple tool for the computation of the bit-security distance between specific distributions that may occur in practice.

To this end, we first show that one can equivalently phrase the study of aborting adversaries in terms of the class of *fuzzy adversaries* $\mathscr{A}_{\approx} := \{\tilde{A} \mid \tilde{A} : \Omega \to [-1, 1]\}$. These adversaries output $y = A(x)$ not only a guess of which distribution they are given (via $y/|y| \in \{\pm 1\}$), but also a *confidence level* $|y| \in [0, 1]$. One then measures the advantage of fuzzy adversaries with a "continuous" analogue of Eq. (2.1) (Definition 15), which we write as $\mathsf{adv}^{\mathsf{MW},\approx}_{\mathscr{X}}(\tilde{A})$. We prove equivalence (Lemma 15) by giving efficient, advantage-preserving transformations between the two classes of adversaries. This shows that, when maximized over the set of all possible adversaries, $\mathsf{adv}^{\mathsf{MW}}(A)$ and $\mathsf{adv}^{\mathsf{MW},\approx}(A)$ are equivalent. Moreover, since the transformations between fuzzy and aborting adversaries used in our proofs also preserve the adversary's running time, they also establish the equivalence between the corresponding notions of *computational* (and, looking forward, *computational-statistical*) bit security.

We then prove a number of useful properties for aborting and fuzzy adversaries. For example, we show that the MW advantage is a convex function of randomized aborting adversary. As a simple corollary, we derive that the optimal advantage is always achieved by a deterministic aborting

adversary (Corollary 1).

Next we dig deeper into the structure of the optimal fuzzy adversary. Here, we show that one may always improve the advantage of a fuzzy adversary by modifying it to have full (or no) confidence on any input (Theorem 5), i.e. optimal fuzzy adversaries have outputs in $\{-1,0,1\}$. Since these values corresponds precisely to the outputs of aborting adversaries $1,\perp,0$, this provides an alternative proof that aborting and fuzzy adversaries are equivalent. More interestingly, we use the lemma to precisely characterize when optimal fuzzy adversaries have full confidence, i.e., output ± 1 instead of 0. Specifically, we show that every optimal fuzzy adversary is of the form

$$A_{\tau}^{\mathscr{X}}(x) = \begin{cases} 0 & \left|\ln \frac{\mathscr{D}_0(x)}{\mathscr{D}_1(x)}\right| < \tau \\ \mathsf{sign}\left(\ln \frac{\mathscr{D}_0(x)}{\mathscr{D}_1(x)}\right) & \text{otherwise} \end{cases},$$

for some threshold $\tau \in [0,\ln(3)]$. Moreover, the value of the optimal threshold τ is uniquely determined as a simple function of the adversary's conditional success probability (Theorem 4).

2.1.2 Watanabe-Yasunaga Bit Security

We next investigate the optimal adversary for Watanabe-Yasunaga Bit Security. On the technical side, our work here is less novel, as information theorists have recently identified [69] a natural choice of adversaries with good performance. Before discussing the precise results, we briefly provide some background. Watanabe-Yasunaga Bit security (as originally defined in [84]) is specified in terms of an "inner" adversary A that on input a sample $x \leftarrow \mathscr{D}_b$, outputs either 0 or 1. This adversary is run n times $y_1 = A(x_1),\ldots,y_n = A(x_n)$ on independent samples $x_i \leftarrow \mathscr{D}_b$ all chosen from the same unknown distribution. The number of samples n is chosen large enough so that the value of the bit b can be determined with very high probability $\mu \approx 1$ (say, $\mu \geq 0.99$) based on the output values $y_1,\ldots,y_n$. So, the total running time is given by $n \cdot T_A$, and [84] defines the bit security to be $\log_2(n \cdot T_A)$, minimized over all inner adversaries A and number of repetitions n such that $\mu \geq 0.99$. They also show that this quantity can be equivalently computed as $\log(T_A/\mathscr{R}_{1/2}(A_0,A_1))$ where $\mathscr{R}_{1/2}$ is the Renyi divergence of order $1/2$, and $A_b = A(\mathscr{D}_b) \in \{0,1\}$ is the Bernoulli random

variable defined by the output of the adversary on input a sample from $\mathscr{D}_b$.

At this point, it is natural to ask:

- What is the relation between the MW and WY bit security?
- What is the optimal adversary $A(x) \in \{0,1\}$ for the WY definition?

Notice that since the WY adversaries always output either 0 or 1, they are potentially easier to use, as the attacker does not have to choose whether or not to abort.

Regarding the first question, [84] proved only the inequality $MW \leq WY$, showing that WY is a more conservative notion of bit security, and leaving a more precise comparison as an open problem. In a follow-up paper [85] the same authors established the equivalence between MW and WY (up to an additive constant), but with a catch. Technically, they prove the equivalence between MW and WY bit security for the same class of aborting adversaries (with output in $\{0,1,\perp\}$ used in [61]. Then, they claim equivalence with the original WY definition by informally stating that the definition in [84] does not explicitly depend on the size of the co-domain of the adversary A. However, the justification is incorrect because the Renyi divergence $\mathscr{R}_{1/2}(A(\mathscr{D}_0),A(\mathscr{D}_1))$ implicitly depends on the size of the output of A. Despite this gap in the proof, we show that the main claim of [85] (about the equivalence of MW and WY bit security) is correct, and in the process we give a simpler proof of this fact.

So, at this point, the WY notion of bit security can be considered an alternative characterization of the MW bit security, rather than a new definition, and the question is whether this alternative characterization can help in evaluating the bit security of decision problems. One seemingly attractive feature of the WY is that it uses standard adversaries $A(x)$ with outputs in $\{0,1\}$. This is because for these adversaries there is a particularly natural attack, that on input a sample x, outputs the bit b for which the probability $\mathscr{D}_b(x)$ is highest. However, this does not seem to help in evaluating the bit security using the WY characterization: we show that there exist distinguishing games where this adversary yields bit security estimates twice as high as the optimal, correct value. Recall that bit security (roughly) measures the *exponent* of the running time of the adversary. So, a multiplicative

factor in bit security estimation is quite large, and the "natural" non-aborting adversarial strategy is far from optimal in the context of the WY definition.

2.1.3 Computational/Statistical Bit Security

Finally, we investigate the definition of (c,s)-bit security proposed in [53], to extend MW bit security to encompass both computational and statistical security. Recall that the MW (computational) bit security of a problem is the highest c such that $T(A)/\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \geq 2^c$ for all adversaries A. Similarly, statistical security can be defined as the smallest s such that $1/\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \geq 2^s$ for all adversaries A, where this time the running time of A is ignored. Li et al. define a protocol to be (c,s)-secure if for any adversary A

$$\text{either} \quad \frac{T(A)}{\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A)} \geq 2^c \qquad \text{or} \quad \frac{1}{\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A)} \geq 2^s.$$

As explained in the introduction, a protocol satisfying this definition seems to provide an adequate level of security whenever computational security and statistical security are considered acceptable individually. Here we point out that a protocol can offer neither c bits of computational security nor s bits of statistical security, and still be (c,s)-secure. Consider for example a protocol such that there exist a very efficient adversary A_c with running time $T(A_c) = 1$ that achieves MW advantage 2^{-s}, and some other adversary A_s with very large running time $T(A_s) \geq 2^c$ that achieves MW advantage ≈ 1. Then, the protocol is neither computationally nor statistically secure because A_c breaks computational security (for $s < c$), and A_s breaks statistical security. So, (c,s)-security is strictly weaker than both c-bits computational security, and s-bits of statistical security. In fact, one should expect this to be the case in any application that makes use of both computational and statistical security primitives, as an adversary can choose to attack the application by trying to break either one or the other type of primitives.

While (c,s)-bit security was introduced in [53][4] (and successfully used to analyze a practical

[4]This work is Chapter 4 of the current document. For ease of readability, we have moved several basic lemmata regarding (c,s)-bit security (and its definition) initially proved in that work into this chapter.

protocol), this was done via direct manipulation of the definition. In this paper we establish a tight connection between the MW advantage $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A)$ and a standard distance measure used in statistics: the (squared) *Le Cam distance* $\Delta^2_{\mathsf{LC}}(A(\mathscr{D}_0),A(\mathscr{D}_1))$ between the adversary's output probability distributions. Then, we use this connection to prove several useful properties of the (c,s)-bit security which support two of the most common proof techniques in cryptography:

- The "hybrid argument" (see Theorem 8 for formal statement): consider a sequence of distributions $\mathscr{D}_0,\ldots,\mathscr{D}_k$. If the game defined by any pair of neighboring distributions $(\mathscr{D}_{i-1},\mathscr{D}_i)$ is (c,s)-secure, then the game defined by the extreme distributions $(\mathscr{D}_0,\mathscr{D}_k)$ is also (c',s')-secure, for $c' \approx c - \log k$ and $s' \approx s - \log k$.

- The "distribution replacement" theorem (see Theorem 9 for formal statement): Consider a decision game $(\mathscr{D}_0^{\mathscr{Y}},\mathscr{D}_1^{\mathscr{Y}})$ parametrized by a distribution $\mathscr{Y}$. If distinguishing between $(\mathscr{Y},\mathscr{Y}')$ is (c,s)-secure, and $(\mathscr{D}_0^{\mathscr{Y}},\mathscr{D}_1^{\mathscr{Y}})$ is (c,s)-secure, then $(\mathscr{D}_0^{\mathscr{Y}'},\mathscr{D}_1^{\mathscr{Y}'})$ is also (c',s')-secure, for $c' \approx c$ and $s' \approx s$.

Our results improve or extend previous work. For example, [61] had already proved a hybrid theorem for computational bit security, and hybrid theorems for statistical bit security are essentially a form of (pythagorean) triangle inequality for the associated distance functions between distributions. The novelty here is to establish the validity of hybrid arguments in the more general computational-statistical setting, where each pair of neighboring distributions $(\mathscr{D}_{i-1},\mathscr{D}_i)$ may be neither computationally nor statistically indistinguishable. Distribution replacement theorems for bit security were previously proved in [61, 86], but only for the setting where $(\mathscr{D}_0^{\mathscr{Y}},\mathscr{D}_1^{\mathscr{Y}})$ are computationally close and $(\mathscr{Y},\mathscr{Y}')$ are *statistically* close (either in the max-log[5] or Hellinger distance). Our theorem allows both $(\mathscr{D}_0^{\mathscr{Y}},\mathscr{D}_1^{\mathscr{Y}})$ and $(\mathscr{Y},\mathscr{Y}')$ to be close in the much weaker sense of computational-statistical bit security.

Both types of techniques are cornerstones for the modular analysis of complex cryptographic protocols that combine several cryptographic primitives. Our results support the uniform use of

[5]This distance measure is defined via $\Delta_{\mathsf{ML}}(\mathscr{D}_0,\mathscr{D}_1) := \max_{x\in\Omega}\left|\ln\frac{\mathscr{D}_0(x)}{\mathscr{D}_1(x)}\right|$, though does not feature in our work.

computational-statistical bit-security to analyze both the final application and its building blocks, including neighboring hybrids $(\mathscr{D}_{i-1}, \mathscr{D}_i)$ and probability replacements $(\mathscr{Y}, \mathscr{Y}')$. Moreover, they support the seamless combination of computational and statistical security primitives, while at the same time offering tight security estimates, which, before our work, could only be done either informally or using ad-hoc methods. The connection with the Le Cam metric, which underlies our proofs, is also of independent interest, and may find other applications.

2.2 Preliminaries

We will often use a particular consequence of the Cauchy-Schwarz inequality known as *Bergström's inequality* (also *Titu's lemma*, *Sedrakyan's inequality*, and *Engel's form*).

Lemma 7 (Bergström's Inequality)**.** *For any real numbers $a_1, \ldots, a_n$, and positive reals $b_1, \ldots, b_n$, we have that*

$$\frac{(\sum_{i\in[n]} a_i)^2}{\sum_{i\in[n]} b_i} \leq \sum_{i\in[n]} \frac{a_i^2}{b_i}.$$

Proof. Rearrange the Cauchy-Schwarz inequality to $\frac{\langle c,d\rangle^2}{\|c\|_2^2} \leq \|d\|_2^2$. Let $c_i := \sqrt{b_i}$ and $d_i := a_i/\sqrt{b_i}$. The claim then follows. □

2.2.1 Cryptographic Games

Cryptographic games are defined by one or more randomized, stateful programs $\mathscr{G}$ used by an adversary A to carry out an attack $A^{\mathscr{G}}$. When running $A^{\mathscr{G}}$, the adversary only has black-box access to $\mathscr{G}$, which is used as an oracle. There are two main categories of cryptographic games. In a *search* game, the final output of $A^{\mathscr{G}}$ is determined by $\mathscr{G}$ and indicates if the attack was successful. In this paper we will be concerned with *decision* games, which are defined by a pair of oracles $\mathscr{G} = (\mathscr{G}_0, \mathscr{G}_1)$ with identical interfaces, so that an adversary A may interact with either of them $A^{\mathscr{G}_0}$, $A^{\mathscr{G}_1}$. This time it is A that produces an output at the end of the interaction, typically consisting of a single bit. We write $A(\mathscr{G}_b)$ for the final output of A at the end of the execution. The goal of the adversary is to determine if it is interacting with either $\mathscr{G}_0$ or $\mathscr{G}_1$.

In the simplest, prototypical example $\mathscr{G}_0$ and $\mathscr{G}_1$ are just distributions over a common set Ω, and the only interaction between A and $\mathscr{G}_b$ is to receive a single sample $x \leftarrow \mathscr{G}_b$. For simplicity, the reader may consider this simple case in mind, and we will often refer to $\mathscr{G}_0, \mathscr{G}_1$ as distributions, and denote them by $\mathscr{D}_0, \mathscr{D}_1$. But all of our results hold for more complex games as well.[6] When studying a decision game $\mathscr{G} = (\mathscr{G}_0, \mathscr{G}_1)$, we write $A(\mathscr{G})$ as an abbreviation for the pair of output distributions $(A(\mathscr{G}_0), A(\mathscr{G}_1))$.

Cryptographic protocols can be parametrized by other cryptographic primitives or distributions used as building blocks. So, for example, we may write $P^{\mathscr{Y}}$ for a cryptographic program P that uses a probability distribution $\mathscr{Y}$, and $P^{\mathscr{Y}'}$ for the same program run with a different distribution Y'. Similarly, security games $(\mathscr{G}_0^{\mathscr{Y}}, \mathscr{G}_1^{\mathscr{Y}})$ can be parametrized by $\mathscr{Y}$.

We remark that the running time of an adversary A against a game $\mathscr{G}$ does not include the time required to run $\mathscr{G}$ in the interaction $A(\mathscr{G})$. In other words, we only account for the time taken by A to write its oracle queries and read the answers. We consider adversaries running in strict (e.g. polynomial) time, i.e., we assume that the running time of A in a run $A(\mathscr{G}_b)$ does not depend on how $\mathscr{G}_b$ answers the oracle queries. In particular, A has the same running time in $A(\mathscr{G}_0)$ and $A(\mathscr{G}_1)$. The running time of an adversary A is denoted by T_A or $T(A)$, and may be a function of a security parameter.

In some settings it is useful to define also a notion of running time for the game $\mathscr{G}$. However, it should be clear that the (total) running time of $\mathscr{G}$ in an execution $A(\mathscr{G})$ typically depends on the adversary A.[7] The time taken to run $\mathscr{G}$ in an execution $A(\mathscr{G})$ is denoted $T_{\mathscr{G}}^A$. Then, we can define the running time of a game $\mathscr{G}$ relative to the running time of A as follows.

Definition 9. *The (relative) running time of $\mathscr{G}$ is defined as the maximum*

$$T_{\mathscr{G}} = \sup_A \frac{T_{\mathscr{G}}^A}{T_A}$$

[6] Formally, for any fixed adversary A and arbitrary decision game $(\mathscr{G}_0, \mathscr{G}_1)$, one may consider the distributions $A^{\mathscr{G}_b}$ (for $b \in \{0,1\}$) on the transcripts of the interaction between A and $\mathscr{G}_b$.

[7] This is most obvious when $\mathscr{G}$ is a game where A may issue an arbitrary number of calls to the game oracles.

over all possible adversaries A.

For decision games $(\mathscr{G}_0, \mathscr{G}_1)$ we always assume that $\mathscr{G}_0$ and $\mathscr{G}_1$ have the same running time. Using this definition, the total running time to run $A(\mathscr{G})$ (including both the time for A and for $\mathscr{G}$) can be bounded as

$$T_{A(\mathscr{G})} = T_A + T_{\mathscr{G}}^A \leq T_A \cdot (1 + T_{\mathscr{G}}) .$$

2.2.2 Bit Security

Consider an adversary A against a decision game $\mathscr{G} = (\mathscr{G}_0, \mathscr{G}_1)$, where $A(\mathscr{G}_b)$ may output $0, 1$ or some other values. Throughout the paper we will use the following definitions and notation:

$$\begin{array}{rrcl} \text{(success probability)} & \beta_A & = & \Pr[A(\mathscr{G}_b) = b] \\ \text{(failure probability)} & \bar{\beta}_A & = & \Pr[A(\mathscr{G}_b) = 1 - b] \\ \text{(output probability)} & \alpha_A & = & \beta_A + \bar{\beta}_A \\ \text{(distinguishing gap)} & \delta_A & = & \beta_A - \bar{\beta}_A \end{array}$$

where all probabilities are computed over the random choice of $b \leftarrow \{0,1\}$, and the randomness of $\mathscr{G}_b$ and A. Notice that, α_A equals the probability that the output of A is in $\{0,1\}$. So, for standard adversaries $A \colon \Omega \to \{0,1\}$ that always output a bit $A(x) \in \{0,1\}$, we have $\alpha_A = 1$ and $\delta_A = 2\beta_A - 1 = \Pr\{A(\mathscr{G}_1) = 1\} - \Pr\{A(\mathscr{G}_0) = 1\}$. But we will use the definition of $\beta_A, \bar{\beta}_A, \alpha_A$ and δ_A also for unrestricted adversaries that may output values outside of $\{0,1\}$. It is well-known that, in the case of probability distributions $(\mathscr{D}_0, \mathscr{D}_1)$, the highest possible distinguishing gap equals the statistical distance

$$\Delta_{\mathsf{SD}}(\mathscr{D}_0, \mathscr{D}_1) = \max_A \delta_A \tag{2.2}$$

and it is achieved by a very simple adversary

$$A_{\mathsf{SD}}^{\mathscr{X}}(x) = \begin{cases} 0 & \mathscr{D}_0(x) > \mathscr{D}_1(x) \\ 1 & \mathscr{D}_0(x) < \mathscr{D}_1(x) \end{cases} \tag{2.3}$$

(When $\mathscr{D}_0(x) = \mathscr{D}_1(x)$, the output of A can be chosen arbitrarily without affecting the gap δ). Since Δ_{SD} is the difference between two probabilities, and the maximum over all A is non-negative, we always have $\Delta_{\mathsf{SD}}(\mathscr{D}_0, \mathscr{D}_1) \in [0,1]$.

The MW Bit Security Measure

Micciancio and Walter [61] suggested to use a more general class of adversaries $\mathscr{A}_\perp$, which map an input $x \in \Omega$ to either 0, 1, or a special "don't know" symbol $\perp$, and demonstrated that these adversaries, together with an appropriate notion of advantage, allows one to resolve several theoretical paradoxes related to the definition of a cryptographically meaningful notion of "bit security". The reader is referred to [61] for intuition and justification of this definition.

Definition 10 (MW Advantage)**.** *For any (possibly randomized) MW distinguisher $A\colon \Omega \to \{0,1,\perp\}$ and distinguishing game $\mathscr{D} := (\mathscr{D}_0, \mathscr{D}_1)$ over Ω, the advantage of an adversary A in distinguishing between $\mathscr{D}_0$ and $\mathscr{D}_1$ is*[8]

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{D}}(A) = \frac{\delta_A^2}{\alpha_A} = \frac{(\beta_A - \bar{\beta}_A)^2}{\beta_A + \bar{\beta}_A}$$

The (squared) MW distance between two distributions is

$$\Delta^2_{\mathsf{MW}}(\mathscr{D}_0, \mathscr{D}_1) = \sup_A \mathsf{adv}^{\mathsf{MW}}_{\mathscr{D}}(A) \in [0,1]. \tag{2.4}$$

If we restrict our attention to "non-aborting" adversaries $A \in \mathscr{A}_{0,1}$, we have $\alpha_A = 1$, and $\mathsf{adv}^{\mathsf{MW}}_X(A) = \delta_A^2$ is the square of the distinguishing gap. This immediately gives the following inequality.

Lemma 8. *For any two distributions $\mathscr{D}_0, \mathscr{D}_1$, we have*

$$\Delta_{\mathsf{SD}}(\mathscr{D}_0, \mathscr{D}_1) \le \Delta_{\mathsf{MW}}(\mathscr{D}_0, \mathscr{D}_1).$$

It is also easy to see that the MW distance satisfies the data processing inequality.

[8] This is syntactically different, but perfectly equivalent to the definition given in [61], which defines the advantage as $\alpha_A \cdot (2\beta_A^* - 1)^2$, where $\beta_A^* = \beta_A/\alpha_A$.

Lemma 9 (Data-Processing Inequality)**.** *Let $\mathscr{D}_0, \mathscr{D}_1$ be distributions on Ω. For any (potentially randomized) function $f : \Omega \to \Omega'$, we have that*

$$\Delta_{\mathsf{MW}}(f(\mathscr{D}_0), f(\mathscr{D}_1)) \leq \Delta_{\mathsf{MW}}(\mathscr{D}_0, \mathscr{D}_1).$$

Proof. For any aborting adversary A, define $A^f(x) := A(f(x))$. It is straightforward to see that

$$\Delta^2_{\mathsf{MW}}(f(\mathscr{D}_0), f(\mathscr{D}_1)) = \sup_{A^f} \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^f) \leq \sup_{A} \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) = \Delta^2_{\mathsf{MW}}(\mathscr{D}_0, \mathscr{D}_1).$$

□

We will use the following construction from [61] to transform an aborting adversary $A \in \mathscr{A}_\perp$ to one with only two possible output values.

Lemma 10 ([61, Lemma 1])**.** *For any pair of distributions $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_1)$, aborting adversary $A \colon \Omega \to \{0, 1, \perp\}$, and value $z \in \{0, 1, \perp\}$, the modified adversary*

$$A^z(x) = \textit{if } (A(x) = z) \textit{ then } A^{A(\mathscr{X})}_{\mathsf{SD}}(z) \textit{ else } \perp$$

has advantage

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^z) = \frac{(\Pr_{A(\mathscr{D}_0)}[z] - \Pr_{A(\mathscr{D}_1)}[z])^2}{2(\Pr_{A(\mathscr{D}_0)}[z] + \Pr_{A(\mathscr{D}_1)}[z])}.$$

The WY Bit Security Measure

In [84], an alternative bit security measure was introduced. The definition is parametrized by a "high enough" probability threshold $\mu \approx 1$, but it can be shown that the precise value of μ has only a marginal impact on the definition. An equivalent quantity (without the parameter μ) is also defined in terms of the Renyi divergence of order $1/2$.

Definition 11. *Let $(\mathscr{D}_0, \mathscr{D}_1)$ be distributions on Ω, $\mu \in [0, 1]$, and $\varepsilon_{A,B} := \Pr_{b,\mathbf{x}_b}[B(A^{\otimes k}(\mathbf{x}_b) = b]$,*

where $A : \Omega \to \{0,1\}$, $k \in \mathbb{N}$, *and* $B_k : \{0,1\}^k \to \{0,1\}$. *Define*

$$\mathsf{WY}^{\mu}_{\mathscr{X}}(A) = \min_{k} \min_{B_k} \{\log_2(k \cdot T_A) \mid \varepsilon_{A,B_k} \geq 1-\mu\}, \qquad \mathsf{WY}^{\mu}_{\mathscr{X}} := \min_{A \in \mathscr{A}_{0,1}} \mathsf{WY}^{\mu}_{\mathscr{X}}(A). \tag{2.5}$$

$$\mathsf{WY}_{\mathscr{X}}(A) := \log_2 T(A) + \log_2 \left\lceil \frac{1}{\Delta_{\mathsf{R};1/2}(A(\mathscr{D}_0), A(\mathscr{D}_1))} \right\rceil, \qquad \mathsf{WY}_{\mathscr{X}} := \min_{A \in \mathscr{A}_{0,1}} \mathsf{WY}_{\mathscr{X}}(A). \tag{2.6}$$

We say that two bit security measures are equivalent if they differ by an additive constant factor. While not highlighted as a formal statement in [84], they show that all these measures are essentially equivalent.

Lemma 11 ([84])**.** *For any distinguishing game* $\mathscr{X} := (\mathscr{D}_0, \mathscr{D}_1)$*, for any constants* $\mu \leq \mu'$*, one has that*

$$\left| \mathsf{WY}^{\mu}_{\mathscr{X}} - \mathsf{WY}^{\mu'}_{\mathscr{X}} \right| \leq O(1). \tag{2.7}$$

$$\left| \mathsf{WY}_{\mathscr{X}} - \mathsf{WY}^{\mu}_{\mathscr{X}} \right| \leq O(1), \tag{2.8}$$

Proof. The (stronger) bound

$$\forall A \in \mathscr{A}_{0,1} : \left| \mathsf{WY}^{\mu}_{\mathscr{X}}(A) - \mathsf{WY}^{\mu'}_{\mathscr{X}}(A) \right| \leq \ln(\ln(\frac{1}{4\mu^2})) \leq O(1) \tag{2.9}$$

follows from simple algebraic manipulations of [84, Lemmas 4 and 6], which bound the minimum k in Eq. (2.5) via

$$\frac{\ln(\frac{1}{4\mu})}{\Delta_{\mathsf{R};1/2}(A(\mathscr{D}_0), A(\mathscr{D}_1))} \leq k \leq \left\lceil \frac{\ln(\frac{1}{4\mu^2})}{\Delta_{\mathsf{R};1/2}(A(\mathscr{D}_0), A(\mathscr{D}_1))} \right\rceil. \tag{2.10}$$

Multiplying by T_A and taking logarithms yields nearly matching upper and lower bounds on $\mathsf{WY}^{\mu}_{\mathscr{X}}$, which suffice to establish Eq. (2.9). One then gets the claimed result by minimizing Eq. (2.9) over A. □

Equivalence with the MW bit security is proved in [85], but technically only for aborting adversaries, which we denote $\mathsf{WY}^{\perp}_{\mathscr{X}} = \min_{A \in \mathscr{A}_{\perp}} \mathsf{WY}_{\mathscr{X}}(A)$.

Lemma 12 ([85, Theorems 1 and 2])**.** *For any distinguishing game* $\mathscr{X} := (\mathscr{D}_0, \mathscr{D}_1)$*,*

$$\left| \mathsf{WY}^{\perp}_{\mathscr{X}} - \mathsf{MW}_{\mathscr{X}} \right| \leq O(1).$$

Note that the measure $\mathsf{WY}^{\perp}_{\mathscr{X}}$ is not *a priori* connected to $\mathsf{WY}_{\mathscr{X}}$, as minimizing over a larger set $\mathscr{A}_{\perp}$ may produce smaller values.

Computational/Statistical Bit security

Sometimes, in cryptography, one can achieve a strong notion of security, where no adversary can break a cryptographic function with high probability, regardless of the computational cost incurred by the attack. In the MW bit-security framework, the number of bits of statistical security of a decisional problem $\mathscr{X}$ can be defined as follows.

Definition 12. *A distinguishing game* $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_1)$ *has s bits of statistical security if for every adversary A,* $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \leq 2^{-s}$.

Contrast this with the definition of (computational) bit-security, where the requirement is that $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \leq T(A) \cdot 2^{-c}$. It immediately follows from the definition that any problem achieving s bits of statistical security, also offers s bits of computational security. So, statistical bit-security is a strengthening of computational bit-security. In particular, when combining computational and statistical primitives within a single protocols, one can treat all of them has achieving a given number $c = s$ of computational security bits. However, this is often undesirable in practice because one typically wants to use a higher value of c than for s. In order to combine computational and statistical bit-security analysis in an efficient manner, [53] proposed the following notion of *computational-statistical* bit-security.

Definition 13 ([53])**.** *A distinguishing game* $\mathscr{X}$ *is said to have* (c,s)*-bits of security if for any*

adversary A,

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \leq \max(T(A)2^{-c}, 2^{-s}),$$

i.e., either $c \leq \log_2 \frac{T(A)}{\mathsf{adv}^{\mathsf{MW},\approx}_{\mathscr{X}}(A)}$, *or* $s \leq \log_2 \frac{1}{\mathsf{adv}^{\mathsf{MW},\approx}_{\mathscr{X}}(A)}$.

The notions of computational and statistical security correponds to the following special cases of (c,s)-security:

- A problem has c bits of computational security iff it is (c,∞)-bit secure
- A problem has s bits of computational security iff if is (∞,s)-bit secure.

One may equivalently view (c,s)-bit security via the lens of (a variant of) $(t(\varepsilon),\varepsilon)$ security, a common definition used in the concrete security literature.

Definition 14. *Let* $I \subseteq [0,1]$. *A cryptographic primitive* Π *is said to be* $(t(\varepsilon),\varepsilon)_I$*-secure in an indistinguishability game* $\mathscr{G}$ *if, for any* $\varepsilon \in I$*, any adversary of advantage* ε *has running time at least* $t(\varepsilon)$.

Lemma 13 ([53])**.** *Let* Π *be a cryptographic primitive, and* $\mathscr{G}$ *be an indistinguishability game. Then the following are equivalent*

1. Π *has* (c,s)*-bits of* $\mathscr{G}$*-security, and*
2. Π *is* $(2^c\varepsilon,\varepsilon)_{[2^{-s},1]}$*-secure in* $\mathscr{G}$.

Since any problem offering s bits of statistical security also offers s bits of computational security, (c,s)-bit security is equivalent to $(\max(c,s),s)$-bit security. In other words, one can always assume $c \geq s$. In particular, computational security can be equivalently formulated as (c,c)-bit security, rather than (c,∞).

2.3 Structure and Properties of Optimal MW Adversaries

In this section we characterize the MW adversaries achieving optimal advantage, and prove some useful properties about them. This is done by introducing an alternative, more general, class

of adversaries (which we call "fuzzy" adversaries,) that still achieves the same optimal advantage (and bit security) of standard MW adversaries. We use the added flexibility provided by fuzzy adversaries to investigate optimal adversarial strategies.

The focus in this section is on statistical security, i.e., (∞, s)-bit security. We recall that a decision game $\mathscr{X}$ has s bits of statistical security if for any adversary $A \in \mathscr{A}_\perp$, we have $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \leq 2^{-s}$. So, the study of statistical security reduces to the study of the maximum achievable advantage $\Delta^2_{\mathsf{MW}}(\mathscr{X}) = \sup\{\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \mid A \in \mathscr{A}_\perp\}$.

MW adversaries are generalized as follows. Recall that the output of an MW distinguisher is either a bit $b \in \{0,1\}$, representing a *high confidence* decision between the two distributions, or a special symbol $\perp$ expressing *no confidence*. We generalize this to distinguishers for which the output confidence level can vary continuously from 0 (no confidence) to 1 (high confidence). For this type of distinguisher, it is convenient to map the two values $b \in \{0,1\}$ to a sign

$$\tilde{b} = (-1)^b = (1-2b) = \pm 1 \tag{2.11}$$

so that the output of A can be described by a single number $\sigma \in [-1,1]$, with $\mathsf{sign}(\sigma) = \sigma/|\sigma| = \tilde{b} \in \{\pm 1\}$ representing the decision bit and $|\sigma| \in [0,1]$ the confidence level.[9] We also set $\tilde{\perp} = 0$, so that any MW distinguisher A with output $A(x) = y \in \{0,1,\perp\}$ can be represented by a fuzzy one $\tilde{A}$ with output $\tilde{A}(x) = \tilde{y} \in \{1,-1,0\} \subset [-1,1]$. Notice that this transformation preserves the cost of the adversary $T(\tilde{A}) = T(A)$ as the only difference between the two is the symbol used to encode the final output. We write $\tilde{\mathscr{A}}_\perp = \{\tilde{A} \mid A \in \mathscr{A}_\perp\}$ for the set of aborting adversaries with this alternative output representation.

Definition 15 (Fuzzy Distinguisher)**.** *A fuzzy distinguisher is a (possibly randomized) function* $A\colon \Omega \to [-1,1]$. *For any two distributions* $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_1)$ *over* Ω*, the* advantage *of A in distinguish-*

[9] When $\sigma = 0$, the confidence $|\sigma| = 0$ is zero, and the decision $\mathsf{sign}(\sigma)$ is irrelevant. For concreteness, we define $\mathsf{sign}(0) = 0$.

ing between $\mathscr{D}_0$ and $\mathscr{D}_1$ is $\mathsf{adv}^{\mathsf{MW},\approx}_{\mathscr{X}}(A) = \frac{\tilde{\delta}_A^2}{\tilde{\alpha}_A}$ *where*

$$\tilde{\delta}_A = \mathbb{E}_{b,x_b}[\tilde{b} \cdot A(x_b)] \qquad \textit{and} \qquad \tilde{\alpha}_A := \mathbb{E}_{b,x_b}[|A(x_b)|]$$

are the correlation (between the correct result and the output of A) and expected confidence of A.

The set of all possible fuzzy distinguishers is denoted $\mathscr{A}_{\approx}$, so that $\tilde{\mathscr{A}}_{\perp} \subset \mathscr{A}_{\approx}$. The following lemma shows that Definition 15 generalizes the MW notion of advantage to a wider class of adversaries.

Lemma 14. *For any MW adversary $A \in \mathscr{A}_{\perp}$ and corresponding fuzzy adversary $\tilde{A} \in \mathscr{A}_{\approx}$ we have* $\tilde{\delta}_{\tilde{A}} = \delta_A$, $\tilde{\alpha}_{\tilde{A}} = \alpha_A$, $T(\tilde{A}) = T(A)$ *and* $\mathsf{adv}^{\mathsf{MW},\approx}_{\mathscr{X}}(\tilde{A}) = \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A)$.

Proof. It is easy to check that $\tilde{\delta}_{\tilde{A}} = \delta_A$ and $\tilde{\alpha}_{\tilde{A}} = \alpha_A$ by evaluating the expectations over the set $0, 1, -1$ of all possible values. It follows that $\mathsf{adv}^{\mathsf{MW},\approx}_{\mathscr{X}}(\tilde{A}) = \tilde{\delta}^2_{\tilde{A}}/\tilde{\alpha}_{\tilde{A}} = \delta_A^2/\alpha_A = \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A)$. □

Based on the above lemma, we will use the notation $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) = \tilde{\delta}^2_A/\tilde{\alpha}_A$ and $\mathsf{MW}_{\mathscr{X}}(A) = \log_2(T(A)/\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A))$ for the advantage and bit security of arbitrary fuzzy adversaries $A \in \mathscr{A}_{\approx}$. The previous definitions for aborting adversaries $A \in \mathscr{A}_{\perp}$ are just a special case, under the mapping $A \mapsto \tilde{A}$ from $\mathscr{A}_{\perp}$ to $\tilde{\mathscr{A}}_{\perp} \subset \mathscr{A}_{\approx}$.

2.3.1 Equivalence of Aborting and Fuzzy adversaries

Using fuzzy adversaries, we may define the maximum (statistical) advantage in attacking a decision game $\mathscr{X}$ as

$$\Delta^{2,\approx}_{\mathsf{MW}}(\mathscr{X}) = \sup\{\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \mid A \in \mathscr{A}_{\approx}\},$$

and similarly for bit security

$$\mathsf{MW}_{\approx}(\mathscr{X}) = \inf\{\mathsf{MW}_{\mathscr{X}}(A) \mid A \in \mathscr{A}_{\approx}\}.$$

Since we are optimizing over a larger class of adversaries than $\mathscr{A}_{\perp}$, it immediately follows from the definitions that $\Delta^2_{\mathsf{MW}}(\mathscr{X}) \leq \Delta^{2,\approx}_{\mathsf{MW}}(\mathscr{X})$ and $\mathsf{MW}(\mathscr{X}) \geq \mathsf{MW}_{\approx}(\mathscr{X})$, and in principle these

inequalities could be strict. But, as we will see, this is not the case, i.e., aborting and fuzzy adversaries define precisely the same notion of advantage and bit security for decision games. This is proved using the following transformation.

Lemma 15. *Let* $\mathsf{N}\colon \mathscr{A}_{\approx} \to \mathscr{A}_{\perp}$ *be the transformation*

$$\mathsf{N}[A](x;r) = \begin{cases} \frac{1-\mathsf{sign}(A(x;r))}{2} & \textit{with probability } |A(x;r)| \\ \perp & \textit{otherwise.} \end{cases}$$

Then, for any decision game $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_1)$ *and adversary* $A \in \mathscr{A}_{\approx}$*, we have*

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) = \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(\mathsf{N}[A]).$$

Proof. We have that

$$\begin{aligned}
\delta_{\mathsf{N}[A]} &= \Pr_{b,x_b;r}[\mathsf{N}[A](x_b;r) = b] - \Pr_{b,x_b;r}[\mathsf{N}[A](x_b;r) = 1-b] \\
&= \mathbb{E}_{b,x_b}\left[|A(x_b)| \cdot \Pr\left[\frac{1-\mathsf{sign}(A(x_b))}{2} = b\right] - |A(x_b)| \cdot \Pr\left[\frac{1-\mathsf{sign}(A(x_b))}{2} = 1-b\right]\right] \\
&= \mathbb{E}_{b,x_b}[|A(x_b)| \cdot (\Pr[\mathsf{sign}(A(x_b)) = 1-2b] - \Pr[\mathsf{sign}(A(x_b)) = -(1-2b)])] \\
&= \mathbb{E}_{b,x_b}[|A(x_b)| \cdot (\Pr[(-1)^b \cdot \mathsf{sign}(A(\tilde{x}_b)) = 1] - \Pr[(-1)^b \cdot \mathsf{sign}(A(x_b)) = -1])] \\
&= \mathbb{E}_{b,x_b}[|A(x_b)| \cdot \mathbb{E}[(-1)^b \cdot \mathsf{sign}(A(x_b))]] \\
&= \mathbb{E}_{b,x_b}\left[(-1)^b \cdot A(x_b)\right] = \delta_A,
\end{aligned}$$

and

$$\alpha_{\mathsf{N}[A]} = \Pr_{b,x_b;r}[\mathsf{N}[A](x_b;r) \neq \perp] = \mathbb{E}_{b,x_b}[|A(x_b)|] = \alpha_A.$$

It then follows that $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) = \frac{{\delta_A}^2}{\alpha_A} = \frac{{\delta_{\mathsf{N}[A]}}^2}{\alpha_{\mathsf{N}[A]}} = \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(\mathsf{N}[A])$, i.e. N preserves the advantage. □

Clearly, the transformation N also preserves the complexity of the adversary $T(\mathsf{N}[A]) = T[A]$, as the additional operations performed by $\mathsf{N}[A]$ have negligible cost. It immediately follows that

aborting and fuzzy adversaries are equivalent, both for statistical and computational bit security.

Theorem 2. *Aborting and Fuzzy MW adversaries are equivalent, i.e., they define the same notions of advantage and bit security*

$$\Delta^{2,\approx}_{\mathsf{MW}}(\mathscr{X}) = \Delta^{2}_{\mathsf{MW}}(\mathscr{X})$$
$$\mathsf{MW}_{\approx}(\mathscr{X}) = \mathsf{MW}(\mathscr{X}).$$

Proof. We need to show that $\Delta^{2}_{\mathsf{MW}}(\mathscr{X}) \geq \Delta^{2,\approx}_{\mathsf{MW}}(\mathscr{X})$ and $\mathsf{MW}(\mathscr{X}) \leq \mathsf{MW}_{\approx}(\mathscr{X})$. For any $A \in \mathscr{A}_{\approx}$, the aborting adversary $\mathsf{N}[A] \in \mathscr{A}_{\perp}$ satisfies

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) = \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(\mathsf{N}[A]) \leq \sup_{A'} \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A') = \Delta^{2}_{\mathsf{MW}}(\mathscr{D}_0, \mathscr{D}_1).$$

Therefore, $\Delta^{2,\approx}_{\mathsf{MW}}(\mathscr{X}) = \sup_A \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \leq \Delta^{2}_{\mathsf{MW}}(\mathscr{X})$. A similar argument works for bit security, using the fact that $T(A) = T(\mathsf{N}(A))$. □

2.3.2 Convexity and Determinism

In general, the adversaries achieving the optimal advantage and bit security could be randomized. In the case of statistical advantage, it is easy to make *fuzzy* adversaries deterministic using the following transformation.

Lemma 16. *Let* $\mathsf{F}\colon \mathscr{A}_{\perp} \to \mathscr{A}_{\approx}$ *be the transformation*

$$\mathsf{F}[A](x) = \Pr_r[A(x;r) = 0] - \Pr_r[A(x;r) = 1]$$

Then, for any decision game $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_1)$ *and adversary* $A \in \mathscr{A}_{\perp}$ *we have*

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \leq \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(\mathsf{F}[A]).$$

In particular, the optimal advantage $\Delta^2_{\mathsf{MW}}(\mathscr{X})$ *is achieved by a* deterministic *fuzzy adversary* $A \in \mathscr{A}_{\approx}$.

Proof. We first show that $\delta_{\mathsf{F}[A]} = \delta_A$. We have that

$$\begin{aligned}
\delta_{\mathsf{F}[A]} &= \mathbb{E}_{b,x_b}[\tilde{b} \cdot \mathsf{F}[A](x_b)] \\
&= \mathbb{E}_{b,x_b}[(-1)^b \cdot (\Pr_r[A(x_b;r) = 0] - \Pr_r[A(x_b;r) = 1])] \\
&= \frac{1}{2}\left(\mathbb{E}_{x_1}[(-1) \cdot (\Pr_r[A(x_1;r) = 0] - \Pr_r[A(x_1;r) = 1])]\right) \\
&+ \frac{1}{2}\left(\mathbb{E}_{x_0}[(+1) \cdot (\Pr_r[A(x_0;r) = 0] - \Pr_r[A(x_0;r) = 1])]\right) \\
&= \frac{\mathbb{E}_{x_1}[\Pr_r[A(x_1;r) = 1]] + \mathbb{E}_{x_0}[\Pr_r[A(x_0;r) = 0]]}{2} \\
&- \frac{\mathbb{E}_{x_1}[\Pr_r[A(x_1;r) = 0]] + \mathbb{E}_{x_0}[\Pr_r[A(x_0;r) = 1]]}{2} \\
&= \mathbb{E}_{b,x_b}[\Pr_r[A(x_b;r) = b]] - \mathbb{E}_{b,x_b}[\Pr_r[A(x_b;r) = 1-b]] \\
&= \beta_A - \bar{\beta}_A = \delta_A.
\end{aligned}$$

We next show that $\alpha_{\mathsf{F}[A]} \leq \alpha_A$. We have that

$$\begin{aligned}
\alpha_{\mathsf{F}[A]} &= \mathbb{E}_{b,x_b}[|\mathsf{F}[A](x_b)|] \\
&= \mathbb{E}_{b,x_b}\left[\left|\Pr_r[A(x_b;r) = 0] - \Pr_r[A(x_b;r) = 1]\right|\right] \\
&\leq \mathbb{E}_{b,x_b}\left[\Pr_r[A(x_b;r) = 0] + \Pr_r[A(x_b;r) = 1]\right] \\
&= \alpha_A.
\end{aligned}$$

It follows that $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) = \frac{{\delta_A}^2}{\alpha_A} \leq \frac{{\delta_{\mathsf{F}[A]}}^2}{\alpha_{\mathsf{F}[A]}} = \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(\mathsf{F}[A])$. Finally, if $A \in \mathscr{A}_{\approx}$ is a (possibly randomized) adversary achieving the optimal advantage $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) = \Delta_{\mathsf{MW}}(\mathscr{X})$, an equivalent deterministic adversary can be be obtained as $\mathsf{F}(\mathsf{N}(A))$. □

Notice that the result of the transformation $\mathsf{F}[A]$ is not in general an efficient algorithm, because it requires the computation of the probabilities[10] $\Pr_r[A(x;r) = b]$ for $b = 0, 1$. So, Lemma 16

[10]Naturally, one can approximate these probabilities in a relatively efficient manner by repeatedly running $A(x;r_i)$ on

says nothing about (computational) bit security under deterministic attacks. In this subsection and the next, the focus is on statistical security.

We would like to prove a similar result for aborting adversaries, i.e., show that randomness is not needed even when using adversaries $A \in \tilde{\mathscr{A}}_\perp$ with output in $\{0,1,-1\}$. Note that starting from a deterministic $A \in \mathscr{A}_\approx$ (guaranteed by Lemma 16,) and then computing $\mathsf{N}[A]$ does not work, because the result of N is generally a randomized algorithm.[11] Instead, we will prove the existence of deterministic optimal aborting adversaries using a convexity argument.

For any adversaries $A, B \in \mathscr{A}_\perp$ and $\theta \in [0,1]$, define the convex combination $C = \theta \cdot A + (1-\theta) \cdot B$ as the (randomized) adversary that runs A with probability θ and B with probability $1-\theta$. Notice that the convex combination is taken over the randomness, not the output of the adversaries, so that the result is still in $\mathscr{A}_\perp$.

Theorem 3. *The advantage* $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A)$ *is a convex function of* $A \in \mathscr{A}_\perp$*, i.e., for any two adversaries* $A, B \in \mathscr{A}_\perp$ *and* $\theta \in (0,1)$*, the convex combination* $C = \theta \cdot A + (1-\theta) \cdot B \in \mathscr{A}_\perp$ *satisfies*

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(C) \leq \theta \cdot \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) + (1-\theta) \cdot \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(B).$$

Proof. Using the definition of C, we see that

$$\begin{aligned} \beta_C = \Pr[C(\mathscr{D}_b) = b] &= \theta \cdot \Pr[A(\mathscr{D}_b) = b] + (1-\theta) \cdot \Pr[B(\mathscr{D}_b) = b] \\ &= \theta \cdot \beta_A + (1-\theta) \cdot \beta_B \end{aligned}$$

a given input x and many independent random r_i. However, this would result in a randomized algorithm.

[11] In fact, $\mathsf{N}[A]$ is deterministic only when $A = \mathsf{F}(A')$ for some deterministic $A' \in \mathscr{A}_\perp$. In other words, N produces a deterministic aborting adversary only if we already have a deterministic aborting adversary to start with.

and similarly for $\bar{\beta}_C$, α_C and δ_C. Therefore, by Lemma 7,

$$\begin{aligned}\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(C) = \frac{\delta_C^2}{\alpha_C} &= \frac{(\theta\cdot\delta_A + (1-\theta)\cdot\delta_B)^2}{\theta\cdot\alpha_A + (1-\theta)\cdot\alpha_B} \\ &\leq \theta\cdot\frac{\delta_A^2}{\alpha_A} + (1-\theta)\cdot\frac{\delta_B^2}{\alpha_B} \\ &= \theta\cdot\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) + (1-\theta)\cdot\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(B).\end{aligned}$$

□

An immediate consequence of convexity is that optimal aborting adversaries $A \in \mathscr{A}_\perp$ can be easily derandomized.

Corollary 1. *For any decision game $\mathscr{X}$, and (randomized) adversary $A(x;r)$ achieving the optimal advantage $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) = \Delta^2_{\mathsf{MW}}(\mathscr{X})$, there is a deterministic optimal adversary A_r such that $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A_r) = \Delta^2_{\mathsf{MW}}(\mathscr{X})$ and $T(A_r) = T(A)$.*

Proof. Any randomized adversary $A \in \mathscr{A}_\perp$ can be written as a convex combination $A = \sum_r \Pr[r]\cdot A_r$ of deterministic adversaries $A_r(x) = A(x;r)$ indexed by the randomness r. It follows by Theorem 3 that

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \leq \sum_r \Pr[r]\cdot\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A_r) \leq \max_r \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A_r). \tag{2.12}$$

Choosing the value of r that achieves the maximum gives a deterministic adversary A_r with an advantage which is at least as good at A. Moreover, if we start from an optimal (randomized) adversary A, then we also have $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A_r) \leq \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A)$, and the bounds (2.12) must hold with equality. This is only possible if $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) = \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A_r)$ for all r (such that $\Pr[r] > 0$), and the randomness r can be chosen arbitrarily. □

Note that the above corollary does not say that the adversary achieving the best bit security is deterministic. It is only for adversaries that achieve the highest advantage (regardless of complexity) that randomness does not help to improve the running time. If achieving the optimal advantage is computationally hard, then randomness may help reduce the bit security.

2.3.3 Threshold Adversaries are Optimal

We now show that the optimal fuzzy MW adversary has a very simple structure, that of a "threshold distinguisher".

Definition 16. *For any two distributions $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_1)$ over Ω, and threshold $\tau \geq 0$, define the* threshold distinguisher $A_\tau^{\mathscr{X}} \in \tilde{\mathscr{A}}_\perp$ *as the algorithm that, on input x, computes $\ell(x) = \ln(\mathscr{D}_0(x)/\mathscr{D}_1(x))$ and outputs 0 if $|\ell(x)| < \tau$, and $\mathsf{sign}(\ell(x))$ otherwise.*

Note that $A_\tau^{\mathscr{X}}$ depends on the distributions $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_1)$, and that $A_0^{\mathscr{X}} = A_{\mathsf{SD}}^{\mathscr{X}}$ is the optimal distinguisher when advantage is measured by the statistical distance.

Throughout this section, we will frequently interchange between ordering $x \in \Omega$ according to $\ell(x)$ and according to $B^{\mathscr{X}}(x) := \frac{\mathscr{D}_0(x) - \mathscr{D}_1(x)}{\mathscr{D}_0(x) + \mathscr{D}_1(x)}$. As we show below, this makes no difference.

Lemma 17. *Let Ω be a set, and $\mathscr{D}_0, \mathscr{D}_1$ be any two distributions on this set. Let*

$$f(x) := \frac{x-1}{x+1}.$$

Then for $x \geq -1$, f is non-decreasing, and satisfies

$$f(\exp \ell(x)) = B^{\mathscr{X}}(x).$$

The inverse, $f^{-1}(x) = \frac{1+x}{1-x}$, is non-decreasing for $x \geq 1$. Finally, $f(-x) = -f^{-1}(x)$ and $f(1/x) = -f(x)$, and therefore

$$\{x \mid |\ell(x)| \leq \tau\} = \{x \mid |B^{\mathscr{X}}(x)| \leq f(\exp(\tau))\}$$

Proof. Note that $\exp \ell(x) = \frac{\mathscr{D}_0(x)}{\mathscr{D}_1(x)}$. We have that

$$f(\exp \ell(x)) = \frac{\frac{\mathscr{D}_0(x)}{\mathscr{D}_1(x)} - 1}{\frac{\mathscr{D}_0(x)}{\mathscr{D}_1(x)} + 1} = \frac{\mathscr{D}_0(x) - \mathscr{D}_1(x)}{\mathscr{D}_0(x) + \mathscr{D}_1(x)} = B^{\mathscr{X}}(x),$$

as claimed. The claimed values of $f(-x)$ and $f(1/x)$ are easily verified. Finally, we have that

$$\begin{aligned}\{x \mid |\ell(x)| \le \tau\} &= \{x \mid -\tau \le \ell(x) \le \tau\} \\ &= \{x \mid \exp(-\tau) \le \exp \ell(x) \le \exp(\tau)\} \\ &= \{x \mid f(\exp(-\tau)) \le B^{\mathscr{X}}(x) \le f(\exp(\tau))\} \\ &= \{x \mid -f(\exp(\tau)) \le B^{\mathscr{X}}(x) \le f(\exp(\tau))\} = \{x \mid |B^{\mathscr{X}}(x)| \le \exp(\tau)\}.\end{aligned}$$

□

A priori, $A \in \tilde{\mathscr{A}}_{\perp}$ may, on input $x \in \Omega$, output an arbitrary value $A(x) \in \{-1, 0, 1\}$. We next show that one may always improve A by modifying it to output either 0 or $(-1)^{\arg\max_b(\mathscr{D}_b(x))}$, e.g. either abort, or output the more likely value of b.

Lemma 18. *For any $A \in \mathscr{A}_{\approx}$ and $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_1)$, the modified adversary*

$$|A|(x;r) = |A(x;r)| \cdot \mathsf{sign}(\mathscr{D}_0(x) - \mathscr{D}_1(x))$$

satisfies $\mathsf{adv}^{\mathsf{MW},\approx}_{\mathscr{X}}(A) \le \mathsf{adv}^{\mathsf{MW},\approx}_{\mathscr{X}}(|A|)$. *Moreover, if $A \in \tilde{\mathscr{A}}_{\perp}$, then $|A| \in \tilde{\mathscr{A}}_{\perp}$.*

Proof. It is straightforward to verify that $||A|(x;r)| \le |A(x;r)|$. We also have

$$\begin{aligned}|\delta_A| &= \left|\mathbb{E}_{b,x_b;r}[(-1)^b \cdot A(x;r)]\right| \\ &= \left|\sum_x \mathbb{E}_r[A(x;r) \cdot \frac{\mathscr{D}_0(x) - \mathscr{D}_1(x)}{2}]\right| \\ &\le \sum_x \mathbb{E}_r[|A(x;r)| \cdot \frac{|\mathscr{D}_0(x) - \mathscr{D}_1(x)|}{2}] \\ &= \sum_x \mathbb{E}_r[|A|(x;r) \cdot \frac{\mathscr{D}_0(x) - \mathscr{D}_1(x)}{2}] \\ &= \left|\mathbb{E}_{b,x_b;r}[(-1)^b \cdot |A|(x;r)]\right| \\ &= \delta_{|A|}.\end{aligned}$$

It follows that $\mathsf{adv}^{\mathsf{MW},\approx}_{\mathscr{X}}(A) = \delta_A^2/\alpha_A \leq \delta_{|A|}^2/\alpha_{|A|} = \mathsf{adv}^{\mathsf{MW},\approx}_{\mathscr{X}}(|A|)$. Finally, it is immediate to verify that if $A(x;r) \in \{0,1,-1\}$ then $|A|(x;r) \in \{0,1,-1\}$. □

Note that adversaries of the form $|A| \in \tilde{\mathscr{A}}_\perp$ are completely parameterized by the choice of $\mathbf{p} \in [0,1]^{|\Omega|}$, where $\mathbf{p}_x = \Pr_r[|A(x;r)| \neq 0]$ is the probability of not aborting on input x. Optimizing the Micciancio-Walter advantage over such adversaries may be phrased as a standard linear programming problem.

Definition 17 (Continuous Knapsack Problem)**.** *Let $W \geq 0$ be the* capacity *of a knapsack, and let $n \in \mathbb{N}$. For $i \in [n]$, let $\mathbf{w}_i$ be the* weight *of the ith material, and $\mathbf{v}_i$ be the* value *of the ith material. The continuous knapsack problem is to choose $\mathbf{x} \in [0,1]^n$ that maximizes $\sum_i x_i v_i$, subject to the constraint*

$$\sum_i x_i w_i \leq W. \tag{2.13}$$

For weights and values that are non-negative, an optimal solution to the continuous knapsack problem will always achieve the weight capacity $\sum_i x_i w_i = W$.

Lemma 19. *For any two distributions $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_1)$, and fixed non-aborting probability $\alpha \in [0,1]$, we have that*

$$A^* = \arg\max_{\substack{A \in \tilde{\mathscr{A}}_\perp \\ \alpha_A = \alpha}} \mathsf{adv}^{\mathsf{MW},\approx}_{\mathscr{X}}(A)$$

is a maximizer of the continuous knapsack problem with capacity α, weights $w_x := \frac{\mathscr{D}_0(x)+\mathscr{D}_1(x)}{2}$, and values $v_x := \left|\frac{\mathscr{D}_0(x)-\mathscr{D}_1(x)}{2}\right|$.

Proof. We have that

$$\arg\max_{\substack{A \in \tilde{\mathscr{A}}_\perp \\ \alpha_A = \alpha}} \mathsf{adv}^{\mathsf{MW},\approx}_{\mathscr{X}}(A) = \arg\max_{\substack{A \in \tilde{\mathscr{A}}_\perp \\ \alpha_A = \alpha}} \frac{\delta_A^2}{\alpha_A} = \arg\max_{\substack{A \in \tilde{\mathscr{A}}_\perp \\ \alpha_A = \alpha}} |\delta_A|.$$

By Lemma 18, we may without loss of generality reduce analyzing adversaries of the form $|A|$. Such adversaries may be completely specified by a vector of probabilities $\mathbf{p} \in [0,1]^{|\Omega|}$ to output 0.

For such adversaries, one can compute that

$$|\delta_A| = \mathbb{E}_{b,x_b;r}[(-1)^b \cdot A(x_b;r)] = \sum_x \mathbb{E}_r[A(x;b)] \frac{\mathscr{D}_0(x) - \mathscr{D}_1(x)}{2} = \sum_x \mathbf{p}_x \left| \frac{\mathscr{D}_0(x) - \mathscr{D}_1(x)}{2} \right|.$$

We are optimizing over the space of all adversaries with non-abort probability α. This constraint may be written as

$$\alpha_A = \alpha \iff \mathbb{E}_{b,x_b;r}[|A(x_b;r)|] = \alpha \iff \sum_x \mathbf{p}_x \frac{\mathscr{D}_0(x) + \mathscr{D}_1(x)}{2} = \alpha.$$

So, the optimal adversary is exactly the optimizer of an instance of the continuous knapsack problem. □

It is well known that the optimal solution to the continuous knapsack problem has a particularly simple form (which may be computed with a standard greedy algorithm).

Lemma 20. *Let $(W, \mathbf{v}, \mathbf{w})$ be an instance of the continuous knapsack problem. Let*

$$\mathbf{x}_i = \begin{pmatrix} 1 & \frac{v_i}{w_i} > \tau \\ \varepsilon & \frac{v_i}{w_i} = \tau \\ 0 & \frac{v_i}{w_i} < \tau \end{pmatrix} \tag{2.14}$$

for some threshold $\tau \geq 0$ and $\varepsilon \in [0,1]$ such that $\sum_{i:\frac{v_i}{w_i}>\tau} w_i + \varepsilon \sum_{i:\frac{v_i}{w_i}=\tau} w_i = W$. Then $\mathbf{x}$ is has at least as high of value among all solutions of the aforementioned continuous knapsack problem instance.

One may modify $\mathbf{x}$ to obtain another optimal solution $\mathbf{x}'$ on the set $\{i \mid \frac{v_i}{w_i} = \tau\}$ while preserving the capacity $\sum_i \mathbf{x}' w_i = W$ and value $\sum_i x'_i v_i$, so $\mathbf{x}$ is not unique, though this will not matter in our work.

Theorem 4. *Let $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_1)$ be a distinguishing game on a finite set Ω. Then, the optimal distinguisher $A \in \tilde{\mathscr{A}}_\perp$ is threshold.*

Proof. By Lemma 19, we have that (for fixed α) a maximizer of the Micciancio-Walter advantage is a maximizer of a particular continuous knapsack instance, namely with capacity α, weights $w_x := \frac{\mathscr{D}_0(x)+\mathscr{D}_1(x)}{2}$, and values $v_x := \left|\frac{\mathscr{D}_0(x)-\mathscr{D}_1(x)}{2}\right|$. By Lemma 20, a maximizer of this instance is given by $\mathbf{x}$ according to Eq. (2.14). Rephrased in terms of Micciancio-Walter adversaries, we have that the optimal adversary takes the form

$$|A^*|(x) = \begin{cases} (-1)^{\arg\max_b \mathscr{D}_b(x)} & \left|B^{\mathscr{X}}(x)\right| > \tau \\ \varepsilon & \left|B^{\mathscr{X}}(x)\right| = \tau \\ 0 & \left|B^{\mathscr{X}}(x)\right| < \tau, \end{cases},$$

for some $\tau, \varepsilon \in [0,1]$, where by outputting ε we mean that $\Pr[|A^*|(x;r) \neq 0] = \varepsilon$ for such x. By Lemma 17, we have that this may be equivalently written as thresholding $|\ln \ell(x)|$ with the threshold $\tau' := \ln \frac{1+\tau}{1-\tau}$. Finally, $|A^*|$ is still randomized (unless $\varepsilon \in \{0,1\}$). But, $|A^*|$ can be written as a convex combination of two deterministic threshold adversaries, so by convexity of the Micciancio-Walter advantage, we have that a deterministic threshold adversary has at least as good advantage.

The above argument shows that for any fixed α, a deterministic threshold adversary is optimal. Optimizing over $\alpha \in [0,1]$, we get optimality of deterministic threshold adversaries in general. □

Finally, we show that an optimal Micciancio-Walter adversary is threshold of *bounded* threshold τ.

Theorem 5. *Let $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_1)$ be a distinguishing game on a finite set Ω. Then*

$$\Delta^2_{\mathsf{MW}}(\mathscr{D}_0, \mathscr{D}_1) = \sup_{\tau \geq 0} \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^{\mathscr{X}}_\tau). \tag{2.15}$$

Moreover, if $\beta^*_A := \beta_A/\alpha_A$ is the conditional success probability of this adversary, then the threshold τ satisfies

$$\tau = \ln\left(\frac{1+3\beta^*_A}{3-2\beta^*_A}\right) \leq \ln(3).$$

Proof. Eq. (2.15) follows immediately from Theorem 4, so we focus on the rest of the result.

Let $T = \{|\ell(x)| \mid x \in \Omega\}$ be the (finite) set of possible thresholds, of order $k \leq |\Omega|$. Order these thresholds $\tau_1 < \tau_2 < \cdots < \tau_k$ in increasing order. As shorthand, write $A_i := A^{\mathscr{X}}_{\tau_i}$.

Fix $i \in [k]$, and consider under what conditions we have that A_{i+1} has advantage better than A_i. Let $\varepsilon \in [0,1]$, and define $A^\varepsilon := \varepsilon \cdot A_{i+1} + (1-\varepsilon) \cdot A_i$. Note that A^ε is randomized, though by convexity of the Micciancio-Walter advantage A^ε has advantage maximized at $\varepsilon \in \{0,1\}$.

We proceed by $\partial_\varepsilon \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^\varepsilon)$, and using this to characterize when $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^{\varepsilon'}) > \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^0)$. Note that by convexity of the Micciancio-Walter advantage, this immediately implies that $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^1) > \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^0)$, e.g. the threshold τ_{i+1} yields better performance than the threshold τ_i.

Note that

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^\varepsilon) = \frac{(\delta_A + \varepsilon\Delta)^2}{\alpha_A + \varepsilon p}.$$

where $\Delta = \sum_{x:|\ln(\ell(x))|=\tau_{i+1}} \left|\frac{\mathscr{D}_0(x^*) - \mathscr{D}_1(x^*)}{2}\right|$, and $p = \sum_{x:|\ln \ell(x)|=\tau_{i+1}} \frac{\mathscr{D}_0(x^*)+\mathscr{D}_1(x^*)}{2}$. One can check that for this p, q, we have that

$$\frac{\Delta}{p} = \left|B^{\mathscr{X}}(x^*)\right|,$$

where x^* is any point such that $|\ell(x)| = \tau_{i+1}$. By Lemma 17, we may compute that $\left|B^{\mathscr{X}}(x^*)\right| = \frac{1+\exp(\tau_{i+1})}{1-\exp(\tau_{i+1})}$. We can then compute

$$\partial_\varepsilon \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A_\varepsilon) = \frac{2\Delta(\alpha_A + \varepsilon p)(\delta_A + \varepsilon\Delta) - p(\delta_A + \varepsilon\Delta)^2}{(\alpha_A + \varepsilon p)^2}.$$

If the numerator is strictly positive at $\varepsilon = 0$, it implies there is sufficiently small $\varepsilon' > 0$ such that $A_{\varepsilon'}$ has advantage better than A_0. This occurs when

$$\frac{\Delta}{p} = \left|B^{\mathscr{X}}(x^*)\right| > \frac{1}{2}\frac{\delta_A}{\alpha_A}.$$

By Lemma 17, we may equivalently write this as

$$|\ell(x)| > \ln f^{-1}(\frac{\delta_A}{2\alpha_A}) = \ln \frac{1+\frac{\delta_A}{2\alpha_A}}{1-\frac{\delta_A}{2\alpha_A}} = \ln \frac{2\alpha_A+\delta_A}{2\alpha_A-\delta_A}.$$

We next note that $\delta_A = \beta_A - \bar{\beta}_A$, and $\alpha_A = \beta_A + \bar{\beta}_A$. We may moreover write $\beta_A^* = \beta_A/\alpha_A$ as the *conditional* success probability. In terms of this quantity, we can write $\bar{\beta}_A = \alpha_A(1-\beta_A^*)$. Combined, this gives us the condition

$$|\ell(x)| > \ln \frac{3\beta_A - \bar{\beta}_A}{\beta_A + 3\bar{\beta}_A} = \ln \frac{1+2\beta_A^*}{3-2\beta_A^*}.$$

As for optimal A we have $\beta_A^* \in [1/2, 1]$, this implies that $\tau_{i+1} \in [0, \ln(3)]$, as claimed. □

2.4 Equivalence of MW and WY bit security

In [85], it is claimed that for any decisional game $\mathscr{X}$, the quantities $\mathsf{WY}(\mathscr{X})$ and $\mathsf{MW}(\mathscr{X})$ are equal up to an additive constant, i.e., the MW and WY notions of bit-security are equivalent. However, [85] only proves the statement for a variant of the WY security definition that uses aborting adversaries (i.e., the MW adversaries with output in $\{0,1,\perp\}$ introduced in [61]), rather than the traditional (non-aborting, inner) adversaries used in [84] to define WY security. To close this gap, [85] informally states that changing the class of adversaries does not affect the definition of $\mathsf{WY}(\mathscr{X})$, and justifies the assertion saying that the definition does not *explicitly* depend on the size of the codomain of the adversary A. However, this reasoning is incorrect because the quantity $\Delta_{\mathsf{R};1/2}(A(\mathscr{X}))$ used in the definition *implicitly* depends on the size of the codomain of A[12]. Still, the main claim in [85] is correct, as shown in the following theorem which gives a direct proof of the equivalence of $\mathsf{WY}_{\mathscr{X}}$ and $\mathsf{MW}_{\mathscr{X}}$.

The theorem makes use of the following technical lemma to modify an aborting adversary in such a way that it uses only two of the output symbols in $\{0,1,\perp\}$.

[12]For large codomains, say growing linearly with the sample complexity to reliably distinguish $\mathscr{X} = (\mathscr{X}_0, \mathscr{X}_1)$, one may appeal to work in communication-constrained binary hypothesis testing [69] to obtain an explicit separation. We omit details for brevity, and point to this work for justification that reliably distinguishing $\mathscr{X} = (\mathscr{X}_0, \mathscr{X}_1)$ and $A(\mathscr{X}) = (A(\mathscr{X}_0), A(\mathscr{X}_1))$ may require different numbers of samples.

Lemma 21. *For any decision game $\mathscr{X}$, and aborting adversary $A \in \mathscr{A}_\perp$, there exists a modified adversary $A' \in \mathscr{A}_\perp$ with output in $\{b, \perp\}$ (for some fixed $b \in \{0,1\}$) and similar running time $T(A) \approx T(A')$, such that*

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \leq 2 \cdot \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A').$$

Proof. Let $A' = A^z$ be the modified adversary from Lemma 10 with z the values in $\{0,1\}$ that maximizes the advantage $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^z)$. For $i \in \{0,1\}, j \in \{0,1,\perp\}$, let $p_{i,j} = \Pr_{A(\mathscr{D}_i)}[j]$, so that $\beta_A = (p_{0,0} + p_{1,1})/2$, $\bar{\beta}_A = (p_{0,1} + p_{1,0})/2$ and, by Lemma 10,

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^j) = \frac{(p_{0,j} - p_{1,j})^2}{2(p_{0,j} + p_{1,j})}.$$

We can then bound

$$\begin{aligned}
\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) &= \frac{(\beta_A - \bar{\beta}_A)^2}{\beta_A + \bar{\beta}_A} \\
&= \frac{1}{2} \frac{((p_{0,0} - p_{1,0}) - (p_{0,1} - p_{1,1}))^2}{(p_{0,0} + p_{1,0}) + (p_{0,1} + p_{1,1})} \\
&\leq \frac{(p_{0,0} - p_{1,0})^2}{2(p_{0,0} + p_{1,0})} + \frac{(p_{0,1} - p_{1,1})^2}{2(p_{0,1} + p_{1,1})} \\
&= \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^0) + \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^1) \\
&\leq 2\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^z)
\end{aligned}$$

where the first inequality is Lemma 7, and the second one follows by our choice of z. □

Theorem 6. *For any decision game $\mathscr{X}$, $\mathsf{WY}(\mathscr{X}) = \mathsf{MW}(\mathscr{X}) + \Theta(1)$.*

Proof. The inequality $\mathsf{MW}_{\mathscr{X}} \leq \mathsf{WY}_{\mathscr{X}}$ was already proved in [84]. Here we prove $\mathsf{WY}_{\mathscr{X}} \leq \mathsf{MW}_{\mathscr{X}} + O(1)$. Let $z \in \{0,1\}$ be the value maximizing the advantage $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^z)$ of the modified adversary defined in Lemma 10. Note that A^z has codomain $\{b, \perp\}$ (rather than $\{0,1\}$). But since $\Delta_{\mathsf{R};1/2}$ does not give any special meaning to the output of the adversary, we can view A^z as a valid

adversary for $\mathsf{WY}_{\mathscr{X}}$. Then, using Lemma 21, we get

$$\begin{aligned}\mathsf{MW}_{\mathscr{X}}(A) &= \log_2 \frac{T(A)}{\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A)} \\ &\geq \log_2 \frac{T(A)}{2\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^z)} \\ &\geq \log_2 \frac{T(A)}{8\Delta_{\mathsf{R};1/2}(A^z(\mathscr{X}))}.\end{aligned}$$

Note that $T(A^z) = T(A) + O(1)$, so we get that $\mathsf{MW}_{\mathscr{X}}(A) \geq \mathsf{WY}_{\mathscr{X}}(A^z) - 3$. □

The previous theorem shows that one can use $\mathsf{WY}(\mathscr{X})$ as an alternative characterization of $\mathsf{MW}(\mathscr{X})$. This is potentially interesting, as $\mathsf{WY}(\mathscr{X})$ only makes use of traditional (nonaborting) adversaries, which are perhaps more intuitive and easier to use. (This was indeed one of the motivations of [84]). In particular, it is tempting to assume that, since the inner adversary of [84] always outputs either 0 or 1 (i.e., it never aborts), the optimal advantage is achieved by the maximum likelihood distinguisher $A^{\mathscr{X}}_{\mathsf{SD}}$. Perhaps counterintuitively, the following theorem shows that this is not the case, and even if [84] does not make use of aborts, the obvious (inner) distinguishing strategy $A^{\mathscr{X}}_{\mathsf{SD}}$ is not optimal, and can in fact substantially overestimate the number of bits of security by a factor[13] close to 2.

Theorem 7. *There exist (efficiently samplable, efficiently computable) distributions $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_1)$ such that*

$$\mathsf{WY}_{\mathscr{X}}(A^{\mathscr{X}}_{\mathsf{SD}}) \geq 2 \cdot \mathsf{MW}(\mathscr{X}) - O(1).$$

Proof. The choice of $\mathscr{X}$ below is from [82, Lemma 2], where it was used to show the sub-optimality of distinguishing a product distribution $\mathscr{X}^{\otimes n} = (\mathscr{D}_0^{\otimes n}, \mathscr{D}_1^{\otimes n})$ by first computing $A^{\mathscr{X}}_{\mathsf{SD}}$ "coordinate-wise" (sometimes called *Scheffé's test*). Consider the distributions $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_1)$ shown

[13] This is a doubling of the number of security bits k, so it corresponds to overestimating the cost of the attack by an exponential factor 2^k.

in following table, where $\varepsilon \leq 1/8$:

	0	1	2
$\mathscr{D}_0$	0.5	$0.5-\varepsilon$	ε
$\mathscr{D}_1$	$0.5-\varepsilon$	$0.5+\varepsilon$	0
$A_{\mathsf{SD}}^{\mathscr{X}}$	0	1	0
$A_{\mathsf{MW}}^{\mathscr{X}}$	$\perp$	$\perp$	0
$A_{\mathsf{SD}}^{\mathscr{X}}(\mathscr{D}_0)$	$0.5+\varepsilon$	$0.5-\varepsilon$	
$A_{\mathsf{SD}}^{\mathscr{X}}(\mathscr{D}_1)$	$0.5-\varepsilon$	$0.5+\varepsilon$	

The table also shows the optimal $A_{\mathsf{SD}}^{\mathscr{X}}$ distinguisher, its output distribution on input $\mathscr{D}_0$ and $\mathscr{D}_1$, and a candidate[14] MW distinguisher which we will use in our proof. The intuition is clear: if the sample is 2, then it certainly comes from distribution $\mathscr{D}_0$, but for the other samples the distinguisher does not have enough confidence to make the call. This distinguisher succeeds with probability $\beta = \varepsilon/2$, but it never fails. So, it achieves advantage $(\beta-\bar{\beta})^2/(\beta+\bar{\beta}) = \beta = \varepsilon/2$. Since A_{MW} runs in constant time, the decisional problem $\mathscr{X}$ has at most $\log_2(2/\varepsilon) = 1+\log_2(1/\varepsilon)$ bits of security.

Let's now estimate the advantage achieved by A_{SD} as an inner distinguisher. We first evaluate the Hellinger distance

$$\Delta_{\mathsf{H}}^2(A_{\mathsf{SD}}^{\mathscr{X}}(\mathscr{D}_0), A_{\mathsf{SD}}^{\mathscr{X}}(\mathscr{D}_1)) = 1-\sqrt{1-4\varepsilon^2} \leq 4\varepsilon^2$$

where we have used the inequality $1-\sqrt{1-x} \leq x$, which is valid for all $x \in [0,1]$. Finally, using Lemma 4, we bound

$$\Delta_{\mathsf{R};1/2}(A_{\mathsf{SD}}^{\mathscr{X}}(\mathscr{X})) \leq 4\Delta_{\mathsf{H}}^2(A_{\mathsf{SD}}^{\mathscr{X}}(\mathscr{X})) \leq 16\varepsilon^2.$$

Since A_{SD} also runs in constant time, the upper bound on bit security it gives is $\log_2(1/(16\varepsilon^2)) = 2\log_2(1/\varepsilon)-4$. □

[14]This is indeed the optimal MW distinguisher when $\varepsilon \leq 1/8$. When $\varepsilon \geq 1/8$, then $A_{\mathsf{SD}}^{\mathscr{X}}$ becomes optimal.

In summary, if $\varepsilon = 2^{-k}$ (for any $k \geq 3$), the bit security is at most $k+1$, but the WY framework with nonaborting distinguisher A_{SD} only provides a very weak bound of $2k-4$.

2.5 A Toolbox for Analysis of (c,s)-Bit Security

We will need a variant of Lemma 21 which gives a tight connection between the MW advantage and the (squared) Le Cam distance of the adversary output probability distributions $A(\mathscr{X})$. A similar statement was previously proved in [85] under the condition that $\Delta_{\mathsf{R};1/2}(A(\mathscr{X})) \leq 1$, and with worse multiplicative constants.

Lemma 22. *For any decision game $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_1)$ and aborting adversary $A \in \mathscr{A}_\perp$, there is a modified adversary $A' \in \mathscr{A}_\perp$ with similar running time $T(A) \approx T(A')$, such that*

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \leq \Delta^2_{\mathsf{LC}}(A(\mathscr{X})) \leq 3\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A').$$

Proof. The proof proceeds as in Lemma 21, using the same notation, except that this time $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^z)$ is maximized over $z \in \{0,1,\perp\}$. As in the proof of Lemma 21, we still have

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \leq \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^0) + \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^1).$$

To prove the new lemma we notice that

$$\Delta^2_{\mathsf{LC}}(A(\mathscr{D}_0), A(\mathscr{D}_1)) = \sum_{j \in \{0,1,\perp\}} \frac{(p_{0,j} - p_{1,j})^2}{2(p_{0,j} + p_{1,j})} = \sum_{j \in \{0,1,\perp\}} \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^j)$$

which is at least $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^0) + \mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^1)$ and at most $3\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^z)$. □

It is easy to bound the advantage of (possibly adaptive) statistical distinguishers.

Lemma 23. *Let $\mathscr{G}$ be the indistinguishability game instantiated with distribution ensembles $\{\mathscr{X}_\theta\}_\theta, \{\mathscr{Y}_\theta\}_\theta$, where $\theta \in \Theta$. Let $q \in \mathbb{N}$. Then, for any (potentially computationally unbounded)*

adversary A making at most q queries to its oracle, we have that

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \leq \frac{q}{2} \max_{\theta \in \Theta} D(\mathscr{X}_\theta || \mathscr{Y}_\theta). \tag{2.16}$$

Proof. View an (adaptive) adversary as post-processing of samples from an arbitrary distribution on query-response pairs $\mathscr{X}_{\hat{\theta}} := ((\hat{\theta}_1, \mathscr{X}_{\hat{\theta}_1}), \ldots, (\hat{\theta}_q, \mathscr{X}_{\hat{\theta}_q}))$ (and similarly for $\mathscr{Y}_{\hat{\theta}}$). We then have that

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A) \leq \Delta^2_{\mathsf{LC}}(\mathscr{X}_{\hat{\theta}}, \mathscr{Y}_{\hat{\theta}}) \leq \frac{1}{2} D(\mathscr{X}_{\hat{\theta}}, \mathscr{Y}_{\hat{\theta}}) \leq \frac{1}{2} \|\widehat{D}(\mathscr{X}_{\hat{\theta}}, \mathscr{Y}_{\hat{\theta}})\|_1 \leq \frac{q}{2} \max_{\theta \in \Theta} D(\mathscr{X}_\theta || \mathscr{Y}_\theta). \tag{2.17}$$

□

Theorem 8. *Let $\mathscr{D}_0, \ldots, \mathscr{D}_k$ be a sequence of cryptographic games. If for all $i = 1, \ldots, k$, $\mathscr{D}_i = (\mathscr{D}_{i-1}, \mathscr{D}_i)$ is (c_i, s_i)-bit secure, then $\mathscr{X} = (\mathscr{D}_0, \mathscr{D}_k)$ is (c, s)-bit secure for*

$$\begin{aligned} c &= \min_i(c_i) - 2\log_2(\sqrt{3}k) \\ s &= \min_i(s_i) - 2\log_2(\sqrt{3}k) \end{aligned}$$

Proof. Using Lemma 22 we get the upper bound

$$\begin{aligned} \sqrt{\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A)} &\leq \Delta_{\mathsf{LC}}(A(\mathscr{D}_0), A(\mathscr{D}_k)) \\ &\leq \sum_i \Delta_{\mathsf{LC}}(A(\mathscr{D}_i), A(\mathscr{D}_{i+1})) \\ &\leq \sqrt{3} \sum_i \max_{z_i} \sqrt{\mathsf{adv}^{\mathsf{MW}}_{\mathscr{D}_i}(A^{z_i})} \\ &\leq \sqrt{3}k \sqrt{\max_i (T(A^{z_i}) 2^{-c_i}, 2^{-s_i})}. \end{aligned}$$

So, since $T(A) \approx T(A^{z_i})$ for all i, the advantage $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A)$ is at most

$$3k^2 \max(T(A) 2^{-\min_i c_i}, 2^{-\min_i s_i}) = \max(T(A) 2^{-(\min_i c_i - 2\log_2(\sqrt{3}k))}, 2^{-(\min_i s_i - 2\log_2(\sqrt{3}k))}).$$

This proves that $\mathscr{X}$ is at least (c, s)-secure. □

This may be seen as an extension of [61, Theorem 7], which is an analogous result for (c,c)-bit security, though with slightly smaller[15] loss of $\log_2(2k^2) = 2\log_2(\sqrt{2}k)$ bits.

We next establish a distribution replacement theorem for (c,s)-bit security for games $\mathscr{G}^{\mathscr{Y}_b}$ parametrized by a distribution $\mathscr{Y}_b$. This was done in [61] under the assumption that $(\mathscr{Y}_0,\mathscr{Y}_1)$ is statistically ((∞,s)-bit) secure, and in [85] under the assumption that $(\mathscr{Y}_0,\mathscr{Y}_1)$ is computationally ((c,c)-bit) secure. We extend this to a (c,s)-bit security assumption below.

Theorem 9. *Let $\mathscr{G},\mathscr{Y}$ be decision games. If $\mathscr{G}^{\mathscr{Y}_0}$ is (c,s)-bit secure, and $\mathscr{Y}$ is (c',s')-bit secure, then $\mathscr{G}^{\mathscr{Y}_1}$ is (c'',s'')-bit secure, where $c'' = \min(c-2, c'-3-\log_2(1+T_{\mathscr{G}}))$, and $s'' = \min(s-2,s'-3)$. In particular, if $\mathscr{Y}$ and $\mathscr{G}^{\mathscr{Y}_0}$ are (c,s)-bit secure and[16] $T_{\mathscr{G}} = O(1)$, then $\mathscr{G}^{\mathscr{Y}_1}$ is almost (c,s)-bit secure, up to a small additive constant term in bit security.*

Proof. Let A be any adversary. By Lemma 22 and the triangle inequality (for Δ_{LC}), we have that

$$\begin{aligned}\sqrt{\mathsf{adv}^{\mathsf{MW}}_{\mathscr{G}^{\mathscr{Y}_1}}(A)} &\le \Delta_{\mathsf{LC}}(A(\mathscr{G}_0^{\mathscr{Y}_1}),A(\mathscr{G}_1^{\mathscr{Y}_1}))\\ &\le \Delta_{\mathsf{LC}}(A(\mathscr{G}_0^{\mathscr{Y}_1}),A(\mathscr{G}_0^{\mathscr{Y}_0})) + \Delta_{\mathsf{LC}}(A(\mathscr{G}_0^{\mathscr{Y}_0}),A(\mathscr{G}_1^{\mathscr{Y}_0})) + \Delta_{\mathsf{LC}}(A(\mathscr{G}_1^{\mathscr{Y}_0}),A(\mathscr{G}_1^{\mathscr{Y}_1})).\end{aligned}$$

We bound each term in the last sum separately. For the middle term, using the upper bound in Lemma 22 and $T(A) = T(A^z)$, we get

$$\Delta_{\mathsf{LC}}(A(\mathscr{G}_0^{\mathscr{Y}_1}),A(\mathscr{G}_0^{\mathscr{Y}_0})) \le \sqrt{3\max_z \mathsf{adv}^{\mathsf{MW}}_{\mathscr{G}^{\mathscr{Y}_0}}(A^z)} \le \sqrt{3\max(T_A 2^{-c}, 2^{-s})}$$

The other terms are bound constructing distinguishers A_0, A_1 against the game $\mathscr{Y}$ as follows. $A_0^{\mathscr{Y}}$ simulates the execution of A in the game $\mathscr{G}_0^{\mathscr{Y}}$ and flips the answer, i.e., it outputs $1-a$ when A outputs $a \in \{0,1\}$, and $\perp$ otherwise. $A_1^{\mathscr{Y}}$ simulates the execution of A in the game $\mathscr{G}_1^{\mathscr{Y}}$, and outputs

[15] One can recover the exact same loss ($\log_2(2k^2) = 2\log_2(\sqrt{2}k)$) by giving a variant of Lemma 22 with constant factor 2 rather than 3. This can be done by comparing $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{Y}}(A)$ to $\Delta^2_{\mathsf{LC}}(\mathscr{D}'_0,\mathscr{D}'_1)$, where $\mathscr{D}'_b \in [0,1]^2$ is the first two coordinates of $A(\mathscr{D}_b) \in [0,1]^3$. This is to say that one can exactly generalize [61, Theorem 7] by working with $(\mathscr{D}'_0,\mathscr{D}'_1)$ that are positive measures of total mass ≤ 1 rather than *probability* measures of total mass $= 1$. We avoid doing this as the quantitative improvement is small, at the cost of a large amount of conceptual overhead.

[16] Recall from Definition 9 that $T_{\mathscr{G}}$ is the relative running time of $\mathscr{G}$. So, $T_{\mathscr{G}} = O(1)$ is quite common, e.g., when oracles calls can be answered in linear time.

the same result as A. Then, we have

$$\begin{aligned}\Delta_{\mathsf{LC}}(A(\mathscr{G}_0^{\mathscr{Y}_1}),A(\mathscr{G}_0^{\mathscr{Y}_0})) &= \Delta_{\mathsf{LC}}(A_0(\mathscr{Y}_0),A_0(\mathscr{Y}_1)) \\ &\leq \sqrt{3\max_z \mathsf{adv}^{\mathsf{MW}}_{\mathscr{Y}}(A_0^z)} \\ &\leq \sqrt{3\max(T(A)(1+T_{\mathscr{G}})2^{-c'},2^{-s'})}\end{aligned}$$

and similarly for the last term $\Delta_{\mathsf{LC}}(A(\mathscr{G}_1^{\mathscr{Y}_0}),A(\mathscr{G}_1^{\mathscr{Y}_0}))$ using adversary A_1. Combining the three terms gives the bound in the theorem. □

2.6 Conclusion and Open Problems

We developed a number of useful tools to evaluate the bit security of decisional cryptographic properties, in the statistical and computational setting, or even combinations of the two. These include a characterization of the structure of the optimal statistical "aborting" adversaries to facilitate the use of approximate probability distributions (like uniform or discrete gaussians), and general hybrid arguments and probability replacement theorems to combine subprotocols together and support modular security analysis. More tools may be added to the toolbox in the future, but we believe that the results presented in this paper already demonstrate that computational-statistical bit-security can be quite usable and useful.

All results in this paper we focused on decisional problems, which are the hardest case, but combining decisional primitives with search ones should be fairly straightforward, as the definion of bit security for search problems is standard. An interesting direction for future work is to explore the space between search and decision problems. These include, for example, problems with small (polynomially sized) search space, like password authenticated key exchange. Two works [61, 49] offer general definitions that interpolate between search and decision problems, but the significance of those definition for intermediate problems is unclear. Similarly to what was done in [61] for search and decision problems, it would be interesting to analyze a representative set of protocols falling in-between search and decision primitives, possibly in conjunction with standard search

and decision primitives, to see if the bit-security estimates provides by those definitions match the cryptographic intuition behind the informal notion of bit-security.

Another interesting direction for further work is to make good use of the definition of computational-statistical bit-security (proposed in [53] and studied in this work) to formally analyze concrete protocols of practical interest, and make provable (still tight) claims about their security.

2.7 Acknowledgments

Chapter 2, in full, is a reprint of the material as it appears in Theory of Cryptography 2024. Micciancio, Daniele; Schultz-Wu, Mark. "Bit Security: Optimal adversaries, Equivalence results, and a Toolbox for Computational/Statistical Security Analysis". The dissertation author was a primary investigator and author of this material.

Chapter 3

Error Correction and Ciphertext Quantization in Lattice Cryptography

3.1 Introduction

Lattice-based cryptography has many advantages over traditional number-theoretic encryption, from conjectured security against quantum attacks, to the ability to perform arbitrary computations over encrypted data, while at the same time enjoying very fast (quasi-linear time) encryption and decryption operations. This is much better than the cubic running time of the modular exponentiation typically used in constructions based on number theory. However, there is one aspect for which lattice-based constructions have always lagged behind number-theoretic ones: key and ciphertext *sizes*. In fact, early proposals of encryption schemes based on lattices suffered from a very poor *rate*, meaning the ratio of the size of a plaintext to the size of a ciphertext was very small.

Improving the rate of encryption schemes is an important and well-studied problem, and a problem with a well-understood solution: hybrid encryption. By using public-key encryption on a fixed size, randomly chosen symmetric key, and then using this key to encrypt the actual message using a much more efficient block cipher, the cost of the public-key operation (both in terms of running time and rate) can be amortized over a large payload. However, by using hybrid encryption one loses one of the main attractions of lattice-based cryptography: the ability to compute on encrypted data, as data is now encrypted using a block cipher with no useful

homomorphic properties. Homomorphically decrypting AES or other "FHE-friendly" block ciphers [3, 4], addresses this problem, but only partially: it allows one to move data from AES (or another symmetric encryption scheme) to lattice-based cryptography and then perform homomorphic computations on it. The reverse step, e.g. converting the FHE ciphertext back to a space-efficient symmetric ciphertext, is an open problem and would seem to require the symmetric cryptosystem to be fully homomorphic. This has motivated the study of lattice-based encryption schemes with better rate, leading to two constructions of lattice-based homomorphic encryption schemes with rate asymptotically close to 1 [9, 33]. In this paper we present a unified study of high-rate lattice-based encryption schemes, presenting a general framework that parameterizes LWE-based (Learning With Error) encryption with two coding-theoretic objects we call *lattice codes*. The simplest lattice-based encryption scheme (originally proposed by Regev [76]), combines an LWE sample with simple scaling and rounding operations. Here, we replace these scalar operations with two arbitrary lattice codes, one used for error-correction (generalizing scaling), and one used for quantization (generalizing rounding). We then show that known constructions of rate $1-o(1)$ encryption [9, 33] can be described as instances of our general constructions for particular choices of lattice codes, and prove upper and lower bounds on the rate achievable in this framework. Analysis of these schemes in our framework highlights inefficiencies in many current constructions, which we fix to attain asymptotic (rate) improvements.

Organization

The rest of this chapter is organized as follows. In the rest of the introduction we provide more details on our technical contributions and related work. In Section 3.2 we present background information on error-correcting codes needed to describe and analyze our construction. In Section 3.3 we present our generalized encryption framework. In Section 3.4 we show how previous constructions can be obtained as special cases of our framework simply by properly choosing a pair of error correcting codes, and also present a construction combining the desirable properties of [33] and [9]. In Section 3.5 we present impossibility results that limit the rate achievable using common subcases of our generalized construction. In Section 3.6, we give concluding thoughts, and present

some open problems.

3.1.1 Our Contributions

There is a well-known strategy for building (private-key) encryption from LWE, namely

- start with an LWE sample $(\mathbf{A}, \mathbf{b} := \mathbf{As} + \mathbf{e})$, and
- add an encoding of the message $\mathsf{encode}(\mathbf{m})$ to the second component.

Provided one can later recover the message $\mathbf{m}$ from the noisy encoding $\mathsf{encode}(\mathbf{m}) + \mathbf{e}$, this suffices to build private-key encryption.

Given the ciphertext $(\mathbf{A}, \mathbf{b} := \mathbf{As} + \mathsf{encode}(\mathbf{m}) + \mathbf{e})$, how might we compress it? The matrix $\mathbf{A}$ is itself uniformly random, and can be easily compressed using standard techniques[1]. Therefore, we focus on compressing $\mathbf{b}$. This is pseudorandom under the LWE assumption, so we must appeal to some form of *lossy* compression. As the ciphertext already contains a form of error-correction, it can plausibly correct some additional noise.

We leverage a form of compression commonly known as *vector quantization*, where one maps a vector $\mathbf{v} \in \mathbb{R}^m$ to some discrete subset, say $\mathbb{Z}^m$, or more generally a lattice. We use this methodology to quantize $\mathbf{b}$ to a nearby lattice point $\lfloor \mathbf{b} \rceil_L \in L$, where $\lfloor \cdot \rceil_L : \mathbb{R}^m \to L$ is a generalized form of rounding, for example by solving the closest vector problem. Provided the sum of the quantization error $[\mathbf{b}]_Q := \mathbf{b} - \lfloor \mathbf{b} \rceil_Q$ and LWE error $\mathbf{e}$ can be corrected by the error-correcting code, our scheme will decrypt correctly, i.e. we will have successfully compressed an LWE ciphertext.

The above describes how our framework leverages two codes E, Q, for error-correction and quantization respectively. Concretely, the quantized LWE encryption scheme using E and Q (which we call $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$) encrypts by computing

$$\mathsf{Enc}_{\mathbf{s}}(\mathbf{m}) := (\mathbf{A}, \lfloor \mathbf{As} + \mathbf{e} + \mathsf{encode}_E(\mathbf{m}) \rceil_Q), \tag{3.1}$$

[1] In theory, the same $\mathbf{A}$ can be reused with many different $\mathbf{s}_i$, making the amortized cost of $\mathbf{A}$ arbitrarily small. In practice, $\mathbf{A}$ is often replaced with a short seed that is deterministically expanded to $\mathbf{A}$. This is process is not fully justified theoretically, but it is easily proved secure in the random oracle model.

where $\mathbf{A} \leftarrow \mathbb{Z}_q^{m \times n}$, and $\mathbf{e} \leftarrow \chi_e^m$ for an error distribution χ_e. This is a mild modification of (standard) LWE-based encryption (see Definition 27 for details). Despite the simplicity of this approach, our framework is

- broad,
- modular, and
- necessary to achieve high rate.

We discuss all of these points next.

Breadth:

Our framework includes all forms of error-correction and vector quantization that are expressible in terms of *lattice codes* (Definition 18), which are the reduction of a q-ary lattice L modulo q. Equivalently, they are discrete subgroups $L_q := (L \bmod q) \subseteq \mathbb{R}^m/q\mathbb{Z}^m$. For any such subgroup, there are (many) fundamental domains V_L such that $L_q + V_L = \mathbb{R}^m/q\mathbb{Z}^m$ is a partition. A lattice code can be thought of as the choice of a pair (L_q, V_L), along with algorithms to efficiently decompose $\mathbb{R}^m/q\mathbb{Z}^m \to (L_q, V_L)$. This includes most techniques of decoding a point $\mathbf{x} \in \mathbb{R}^m$ to $\lfloor x \rceil \in L$, say by solving the closest vector problem exactly, or approximately via techniques such as Babai's Nearest Planes [5].

In Section 3.4, we instantiate our framework with many different non-trivial LWE-based encryption schemes. In particular, we show that all existing rate $1 - o(1)$ encryption schemes [9, 33] fit into our framework. Beside the schemes that we explicitly analyze, our framework additionally includes any scheme that encodes messages into a lattice for error correction (of which there are many [33, 74, 76, 78]). All known cryptosystems which quantize ciphertexts are expressible in our framework, although this is a much shorter list (containing solely [9][2], and schemes which quantize via rounding each coordinate independently, which are common in practice [24, 28]).

Moreover, we demonstrate the ease of working in our framework by "quantizing" several pre-existing cryptosystems. One such construction combines the desirable properties of [9, 33], namely

[2]We defer discussion of how one can realize this work in our framework to Section 3.4.3.

it encodes messages under a "gadget" lattice (similar to [33]), but attains the same (quasi-optimal) rate as [9].

Modular:

Our framework separates the coding-theoretic analysis from the cryptographic analysis of encryption schemes. The cryptographic analysis of schemes in our framework is somewhat basic. We establish in Theorem 11 that $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ is $\mathsf{RND\text{-}CPA}$-secure[3] (but potentially incorrect) for any choice of E, Q via a simple argument.

The coding-theoretic analysis is similarly straightforward. We express the rate of our cryptosystem in terms of a simple function of the LWE modulus q, dimension m, and volumes $\det E$ and $\det Q$ of the fundamental domains of E, Q.

Correctness analysis requires some knowledge about the shape of these fundamental domains, although we find that it is enough to know their packing and covering radii in the ℓ_2 and ℓ_∞ norms. This analysis frequently highlights inefficiencies in the choice E, Q of codes a cryptosystem (implicitly) uses. Most commonly, the quantizer Q can be replaced with a sparser quantizer Q' without (asymptotically) impacting the correctness of the cryptosystem. We make this modification in several cases, and often find asymptotic improvements. We summarize the results of our analysis in Table 3.1. Our optimizations tend to improve constructions from rate $1 - f(m)$ to $1 - \frac{f(m)}{\log_2 m}$, i.e. improve on known constructions by a logarithmic factor in the dimension. We discuss the reason for these small improvements shortly.

Necessary:

Our framework allows us to derive (strong) coding-theoretic bounds on the rate of $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$, for broad classes of E, Q. Our bounds are on the *rate* of $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$, namely we show it can be at most $1 - f(n,q,m,\sigma,\delta)$ for explicit functions $f(\cdot)$ of the scheme parameters. Under the assumption that $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ meets some notion of correctness (described next), we show universal rate bounds of the above form in the settings of

[3] This is a stronger notion of security than $\mathsf{IND\text{-}CPA}$-security, where one requires ciphertexts be pseudorandom, see Definition 25.

Table 3.1. The lattice codes E, Q that parameterize the Quantized Encryption schemes $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ we study in Section 3.4. Here, $\mathbf{g}_p^t = (1, p, p^2, \ldots, p^{\ell-1})$ for $\ell = \lceil \log_p q \rceil$, $\mathbf{u}_m^t = (1,1,\ldots,1)^t \in \mathbb{R}^m$, and k is a free parameter, typically set to some small polynomial in n. Note that the various parameters p, q, m, k may be required to satisfy certain divisibility constraints, see details in Section 3.4. The rates are computed assuming Gaussian parameter $\sigma = \Theta(\sqrt{n})$, secret key length $n = \Theta(m)$, ciphertext dimension m, and decryption failure rate $\delta = \exp(-n)$. The quality of a gadget (defined in [32]) directly controls noise growth of scalar multiplications (and any operations that use scalar multiplication as a sub-routine) in "Gadget-based" FHE constructions, i.e. smaller quality parameter leads to lower noise growth FHE constructions. Note that gadget encryption is also closely related to GSW-based encryption, see [58].

Name	E	Q	Rate	Quality of E	Source
Regev	$(q/p)\mathbb{Z}^m$	$\mathbb{Z}^m$	$1 - O(1)$	N/A	[76]
Quantized Regev	$(q/p)\mathbb{Z}^m$	$k\mathbb{Z}^m$	$1 - O\left(\frac{1}{\log_2 \frac{q}{k}}\right)$	N/A	Cor. 4
GH	$\Lambda_q^{\perp}(\mathbf{g}_p^t) \otimes \mathbb{Z}^{m/\ell}$	$\mathbb{Z}^m$	$1 - O(1)$	$O(q/p)$	[33]
Quantized GH	$\Lambda_q^{\perp}(\mathbf{g}_p^t) \otimes \mathbb{Z}^{m/\ell}$	$k\mathbb{Z}^m$	$1 - O\left(\frac{1}{\log_2 \frac{q}{k}}\right)$	$O(q/p)$	Sec. 3.4
BDGM	$(q/p)\mathbb{Z}^m$	$\Lambda_{q/p}(\mathbf{u}_m^t)$	$1 - O\left(\frac{\log_2(m\sigma)}{m \log_2 p}\right)$	N/A	[9]
Gadget	$\Lambda_q(\mathbf{g}_p^t) \otimes \mathbb{Z}^{m/\ell}$	$\mathbb{Z}^m$	$1 - O(1)$	$O(p)$	[58]
Quantized Gadget	$\Lambda_q(\mathbf{g}_p^t) \otimes \mathbb{Z}^{m/\ell}$	$\Lambda_{q/p}(\mathbf{u}_m^t)$	$1 - O\left(\frac{\log_2(m\sigma)}{m \log_2 p}\right)$	$O(p)$	Cor. 7

- **Trivial Quantization**: Arbitrary E, with $Q = \mathbb{Z}^m$, and
- **Small Quantization**: Arbitrary E, with $\sqrt[m]{\det Q} \leq O(\sigma)$ of the same size as the LWE error.

We investigate two correctness notions, namely

- **Bounded Noise**: decryption failure rate $\delta = 0$, with respect to bounded noise of the same size (with high probability) as Gaussian noise of parameter σ (in an ℓ_2 ball of radius $\sqrt{m}\sigma$), and
- **Unbounded Noise**: decryption failure rate $\delta > 0$, with respect to arbitrary (concentrated) noise of variance σ^2.

For the first correctness notion, we proceed via "packing bounds", while in the second we proceed via "anti-concentration bounds". Throughout, we state the interesting consequences of our bounds for the case of q polynomially large, see Section 3.5 for full statements.

Our first set of bounds are in the bounded noise model. In this setting, the assumption $\delta = 0$ implies that E_q is a *packing* of $\mathbb{R}_q^m$, meaning that for S the support of the noise (either solely the

LWE error, or the sum of the LWE and quantization error), the sets $\{v+S\}_{v\in E_q}$ are all disjoint, i.e. one can always (uniquely) decode the noisy encoded points $v+S$ back to $v \in E_q$.

Under the assumption E_q is a packing, we follow a standard volume-based argument (called the *sphere packing* or *Hamming* bound, depending on the context) to obtain an inequality between our parameters of interest. Instantiating this argument in the setting of trivial quantization $Q = \mathbb{Z}^m$ leads to the following bound (Theorem 12).

Bound 1. *For any lattice code E, $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,\mathbb{Z}^m]$ has rate at most $1-\Omega(1)$, i.e. rate $1-o(1)$ encryption is impossible.*

This rules out the *a priori* appealing possibility of achieving high-rate encryption by solely optimizing over the error-correcting code E, and motivates investigating further techniques (e.g. quantization).

To handle non-trivial quantization, we require a heuristic assumption (Heuristic 1) that the LWE noise and quantization noise are independent, though we can remove this heuristic for a mild modification of our framework (Section 3.3.2). Our next bound (Theorem 13) then proceeds in essentially the same way, albeit in the case of small quantization, where the set S is more complicated.

Bound 2. *Under a heuristic assumption, for any lattice codes E,Q, if there exists $\varepsilon > 0$ such that $\sqrt[m]{\det Q} = \sigma^{1-\varepsilon}$, then $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ has rate $1-\Omega(1)$, i.e. rate $1-o(1)$ encryption is impossible. If instead $\sqrt[m]{\det Q} \le O(\sigma)$, then rate $1-o\left(\frac{1}{\log_2 q}\right)$ encryption is impossible.*

Therefore, in the bounded error model, quantization is necessary to achieve rate $1-o(1)$ encryption from polynomial modulus. One can further show the aforementioned bounds are tight by repeating the analysis of Corollary 4 in this noise model, though we omit this analysis for brevity.

Our remaining bounds are in the more general setting of δ-correct encryption (for $\delta > 0$) with respect to what is known as *log-concave* noise. We include a brief primer on these random variables in Section 3.2.5, but for now simply state they include (continuous variants of) all of the

noise distributions relevant to public-key lattice-based cryptography, and admit anti-concentration bounds of the form we will require.

The anti-concentration techniques yield bounds with more technical caveats (so *weaker* than the bounded noise model), although one of the bounds is "dimension dependent", which we leverage to give a *stronger* bound than any of our results in the bounded noise model.

Recall that to prove correctness of cryptographic constructions, one often upper bounds the decryption failure rate using concentration inequalities. To prove impossibility results in this noise model, we *lower bound* the decryption failure rate using *anti-concentration* inequalities (Proposition 7), i.e. upper bounds (rather than lower) on how likely it is for a random variable to be close to any particular point (such as its mean).

Our first bound is again for the case of no trivial quantization.

Bound 3. *For any lattice code E, either*

- *the rate of* $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,\mathbb{Z}^m]$ *is* $1-\Omega(1)$*, i.e. not rate* $1-o(1)$*, or*
- *the normalized covering radius satisfies* $\overline{R}_E = \Omega(m)$.

While this bound is weaker than its analogue in the bounded noise model, we expect this to be a proof artifact — it would be quite peculiar if the way to achieve rate $1-o(1)$ encryption was to use codes E for error-correction that are very bad *quantizers*[4]. Note that this result does suffice to rule out rate $1-o(1)$ encryption from a class of *a priori* interesting codes (Corollary 9), namely codes E that are nearly optimal for *both* error-correction and quantization. Such codes are known to exist via randomized constructions, and are nearly optimal in many (non-cryptographic) settings.

Our next bound (Theorem 15) again extends our prior bound to the case of $\sqrt[m]{\det Q} \leq O(\sigma)$.

Bound 4. *Under a heuristic assumption, for any lattice codes E, Q with $\sqrt[m]{\det Q} \leq O(\sigma)$, the rate of* $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ *is at most*

$$1-\Omega\left(\frac{1}{m\log_2(q/\sigma)}\right). \tag{3.2}$$

[4]For an indication of how bad $\overline{R}_E = \Omega(m)$ is, the most trivial lattice $\overline{R}_{\mathbb{Z}^m} = \Theta(\sqrt{m})$ is within a constant factor of being an optimal quantizer.

This bound is tight up to the $\log_2(q/\sigma)$ factor. Note that this bound explicitly depends on the dimension m, instead of solely σ, q. This is significant, due to a simple result (Lemma 30) showing that the rates of $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ and $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E\otimes\mathbb{Z}^k, Q\otimes\mathbb{Z}^k]$ are equal[5] for any k. As one can see from Table 3.1, lattices of this form (for large $k = O(m/\log_2 m)$) are incredibly common in practice. All constructions we are aware of (except for [9]) can be instantiated in our framework using lattices of this type. As a result, one gets a refinement of Bound 4 in this exceedingly common setting.

Bound 5. *Under a heuristic assumption, for any lattice codes $E = E' \otimes \mathbb{Z}^{m/\log_2 m}, Q = Q' \otimes \mathbb{Z}^{m/\log_2 m}$ with $\sqrt[m]{\det Q} \leq O(\sigma)$, the rate of $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ is at most*

$$1 - \Omega\left(\frac{1}{(\log_2 m)\log_2(q/\sigma)}\right).$$

While this is still theoretically rate $1 - o(1)$, practically (for cryptographically relevant dimensions) the convergence is slow. This can be readily observed via concrete comparisons (Figure 3.1), where we find a practical gap between cryptosystems that satisfy the preconditions of Bound 5 (all of which are rate ≤ 0.9) and those that do not (of rate ≈ 1).

Fortunately, one can get around this (exponentially) stronger bound by appealing to lattices without this special structure, such as the quantizer $\Lambda_{q/p}(\mathbf{u}^t_m)$ of [9]. As already summarized in Table 3.1, we find that the pre-existing scheme of [9] is within an $O(\log_2 m)$ factor of optimal, i.e. beats Bound 5 by a significant margin. We then reuse the quantizer $\Lambda_{q/p}(\mathbf{u}^t_m)$ to quantize messages encoded with a "gadget" $\Lambda_q(\mathbf{g}^t_p) \otimes \mathbb{Z}^{m/\ell}$ (similarly to [33], though with a different "gadget" that does not require super-polynomial moduli q), while attaining the same (much higher) rate as [9]. We view this construction as simultaneously achieving the best properties of both of [9, 33] at no cost[6].

[5] There is a mild caveat that various parameters q, n, δ, σ may (implicitly) depend on $m = \dim E = \dim Q$, and these must be taken to be the same size for both instantiations. This will not impact the conclusions we draw from this bound.

[6] There may be some poly-logarithmic overhead in encoding and decoding, but in practice this seems small compared to computing the matrix-vector multiplication as part of LWE-based encryption.

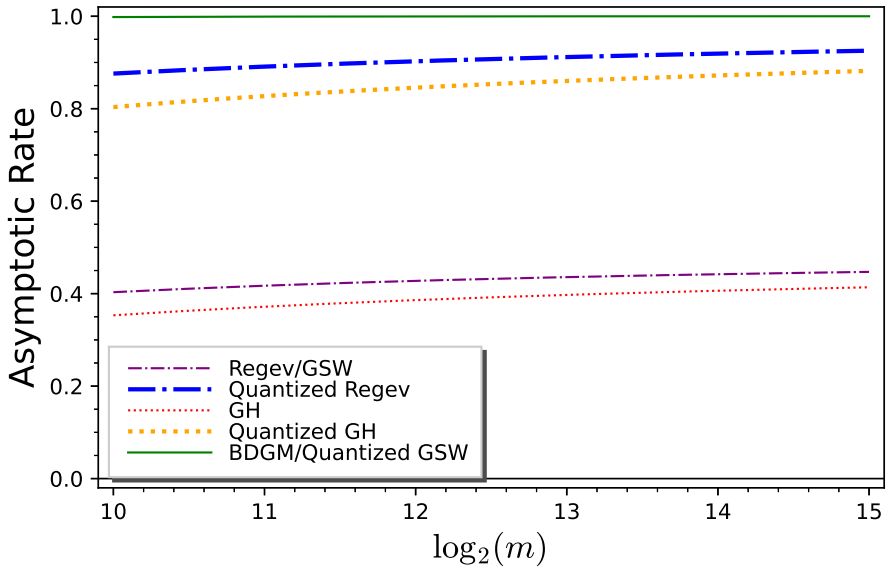

Figure 3.1. The rate of the various cryptosystems $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$, for the codes E,Q in Table 3.1. Throughout, we assume that $q \leq m^2$, $m = n$, $\delta = \exp(-128)$, $\sigma = 2\sqrt{n}$, and then optimize p and k to attain as high rate as possible for $m \in [2^{10}, 2^{15}]$, the range of dimensions included in the Homomorphic Encryption Standard [2].

Optimal Decoding for the Quantizer of [9]:

Independently of the rest of our work, we give an (optimal) $O(m \log_2 q)$ complexity algorithm (Corollary 2) to solve the closest vector problem on the lattice $\Lambda_q(\mathbf{u}^t_m)$, via a simple reduction to a $O(m \log_2 q)$-time CVP algorithm for the scaled root lattice qA^*_{m-1} [56]. We expect this CVP algorithm to be broadly applicable, due to this quantizer leading to constructions that do not satisfy the preconditions of Bound 5. While $\Lambda_q(\mathbf{u}^t_m)$ is used for quantization in [9], a formal decoding algorithm was not given (instead they focused on bounding the ℓ_∞ covering radius of $V_{\Lambda_{q/p}(\mathbf{u}^t_m)}$). From the description in [9], there is an obvious sorting-based algorithm of complexity $\Theta(m(\log_2 m)(\log_2 q))$, i.e. slightly slower than our optimal algorithm. Our algorithm also has the benefit of having a simple to analyze distribution of quantization errors, namely for many distributions of random inputs[7] it is uniform over an explicit convex body[8].

[7] In particular, this holds for what are known as *modulo uniform* distributions, see Chapter 4 of [87].

[8] This is $V_{\Lambda_{q/p}(\mathbf{u}^\ell_m)}$, which by Lemma 25 is the Minkowski sum of a (scaled) permutahedron and an interval.

Log-Concavity of Distributions Relevant to Lattice-based Cryptography

As mentioned before, we leverage the class of *log-concave* random variables. Much of our analysis can be done by simply quoting standard references regarding this topic (for example [80]). To justify the claim that our lower-bounds apply to all noise distributions one encounters in public-key (algebraically-unstructured) lattice-based cryptography, we additionally require that $\langle \mathbf{e}, \mathbf{e}' \rangle$ is log-concave (for $\mathbf{e}, \mathbf{e}'$ independent Gaussians) as well as $\langle \mathbf{e}, \mathbf{e}_K \rangle$ is log-concave (for $\mathbf{e}$ Gaussian, $\mathbf{e}_K$ uniform over a convex body K). We establish these results in Section 3.2.5, though for simplicity of presentation we focus on the case of private-key encryption in the main body of our paper.

3.1.2 Related Work

Our framework is similar to those of [78], which parametrizes the design of lattice-based KEMs via two nested[9] (lattice-based) error-correcting codes. Despite these similarities, [78] does not include bounds on constructions built within their framework, and moreover only considers instantiations with $E = E' \otimes \mathbb{Z}^k \subseteq Q' \otimes \mathbb{Z}^k = Q$ sharing a common low-dimensional structure with $\dim E' = \dim Q' = 8$, which by Bound 5 leads to constructions of severely limited rate.

The framework that has the most similar methods to ours is the framework for the construction of lattice-based KEMs of [43]. They parameterize the construction of lattice-based KEMs via novel primitives they call *Key Consensus* and *Asymmetric Key Consensus (AKC)*, and prove inequalities similar to our rate bounds in this setting. In comparison to our work, they require the assumption of perfect correctness ($\delta = 0$), and solely prove impossibility results in the setting of *single dimension* lattices. This leads them to suggest lattices of the form $Q = Q' \otimes \mathbb{Z}^k$ for $\dim Q' = O(1)$ as "optimal", which (again by Bound 5) is the opposite of what we find.

There is a relatively large body of work that (essentially) quantizes with $Q = c\mathbb{Z}^m$ a scaled integer lattice, dating back to Peikert's work quantizing LWE-based encryption [66], as well as cryptosystems based on the Learning with Rounding problem [7, 24]. Additionally, the "modulus

[9] Note that our framework does not require a nesting assumption.

switching" technique [10, 11] used in the Fully Homomorphic Encryption literature can be viewed from this perspective.

The work of [37] similarly obtains bounds on (public-key) constructions achievable from LWE with polynomially-large modulus, although they show the impossibility of *non-interactive* key exchange, rather than bounds on the rate of constructions.

Finally, our work is closely related to the currently-known rate $1 - o(1)$ lattice-based encryption schemes [9, 33], as a large motivation for our work was to find a way to formally compare the techniques underlying their design.

3.2 Preliminaries

3.2.1 Lattices

A *lattice* is a discrete subgroup $L \subseteq \mathbb{R}^n$. The *rank* of a lattice is the dimension of the $\mathbb{R}$-subspace that it spans. Any rank k lattice can be written as $\mathbf{B}\mathbb{Z}^k$, where $\mathbf{B} \in \mathbb{R}^{n \times k}$ is a *basis* of its linear span. A lattice is called *full-rank* if its rank equals its dimension. Associated with any lattice L is its *dual lattice* $L^* = \{\mathbf{x} \in \mathsf{span}_{\mathbb{R}}(L) \mid \forall \mathbf{v} \in L, \langle \mathbf{x}, \mathbf{v} \rangle \in \mathbb{Z}\}$. The *determinant* of a lattice $L = \mathbf{B}\mathbb{Z}^k$ is the k-dimensional volume of its fundamental region $\mathbf{B}[0,1)^k$. The determinant does not depend on the choice of the basis $\mathbf{B}$, and can be efficiently computed as $\det(L) = \sqrt{\det \mathbf{B}^t\mathbf{B}}$, where $\det \mathbf{B}^t\mathbf{B}$ is the matrix determinant of $\mathbf{B}^t\mathbf{B} \in \mathbb{R}^{k \times k}$.

We say that L is a q-ary lattice if $q\mathbb{Z}^m \subseteq L$, i.e., L is periodic modulo q. Notice that q-ary lattices are always full rank, and the vectors of a q-ary lattice do not necessarily have integer coordinates. There are two standard q-ary integer lattices associated to any matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$:

$$\Lambda_q^{\perp}(\mathbf{A}) = \{\mathbf{x} \in \mathbb{Z}^m \mid \mathbf{A}\mathbf{x} \equiv 0 \bmod q\},$$
$$\Lambda_q(\mathbf{A}) = \{\mathbf{y} \in \mathbb{Z}^m \mid \exists \mathbf{x} \in \mathbb{Z}_q^m \text{ s.t. } \mathbf{A}^t\mathbf{x} = \mathbf{y} \bmod q\}.$$

These lattices are scaled duals of each other, meaning $\Lambda_q(\mathbf{A})^* = \frac{1}{q}\Lambda_q^{\perp}(\mathbf{A})$. For a q-ary lattice, we define the scaled dual as $L^{\perp} = qL^*$, which is such that $\Lambda_q(\mathbf{A})^{\perp} = \Lambda_q^{\perp}(\mathbf{A})$. We say that a

matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ is *primitive* if $\mathbf{A}\mathbb{Z}_q^m = \mathbb{Z}_q^n$, i.e. it is a surjection. For primitive matrices $\mathbf{A}$, $\det(\Lambda_q(\mathbf{A})) = q^{m-n}$ and $\det(\Lambda_q^\perp(\mathbf{A})) = q^n$.

We say that two full-rank lattices L, L' are *nested* if $L' \subseteq L$. Given nested lattices $L' \subseteq L$, the quotient L/L' forms a group of size $\frac{\det L'}{\det L} \in \mathbb{N}$, and therefore $\det(L)$ divides $\det(L')$. Any two lattices $L \subset \mathbb{R}^n$ and $L' \subset \mathbb{R}^{n'}$, can be combined into the *direct sum* $L \oplus L' \subset \mathbb{R}^{n+n'}$, and the *tensor product* $L \otimes L' \subset \mathbb{R}^{n \cdot n'}$. The direct sum is simply the Cartesian product of the two lattices $L \oplus L' = L \times L'$, obtained by concatenating vectors from L and L'. If $\mathbf{A}$ and $\mathbf{B}$ are bases of L and L', then the tensor product $L \otimes L'$ is the lattice with basis $\mathbf{A} \otimes \mathbf{B}$ given by the Kronecker product of $\mathbf{A}$ and $\mathbf{B}$, i.e., the block matrix obtained replacing each entry $a_{i,j}$ of $\mathbf{A}$ with the block $a_{i,j} \cdot \mathbf{B}$. The tensor product $L \otimes L'$ satisfies $\det(L \otimes L') = \det(L')^n \cdot \det(L)^{n'}$. The k-fold direct sum of a lattice $L^{\oplus k} = \oplus_{i=1}^k L$ can be equivalently expressed as the tensor product $L^{\oplus k} = \mathbb{Z}^k \otimes L$.

3.2.2 Convex Bodies

We say a set $K \subseteq \mathbb{R}^n$ is *convex* if, for any $\mathbf{x}, \mathbf{y} \in K$, and $t \in [0,1]$, $(1-t)\mathbf{x} + t\mathbf{y} \in K$. We furthermore say K is *symmetric* if $\mathbf{x} \in K \iff -\mathbf{x} \in K$. Associated with any convex symmetric set K is a *norm* $\|\mathbf{x}\|_K = \inf\{t > 0 \mid \mathbf{x}/t \in K\}$. For such K, we define the *ℓ_p-packing radius* $r_K^{(p)}$ to be the maximal r such that $r \cdot \mathscr{B}_n^{(p)} \subseteq K$. Similarly, we define the *ℓ_p-covering radius* $R_K^{(p)}$ to be the minimal R such that $K \subseteq R \cdot \mathscr{B}_n^{(p)}$. Again, when p is omitted, we mean $p = 2$. For a pair of convex symmetric sets K, K', we write $\|K'\|_K := \sup_{\mathbf{x} \in K'} \|\mathbf{x}\|_K$. We will need the following bounds, which are straightforward to derive.

Lemma 24. *Let K, K' be convex symmetric sets in $\mathbb{R}^n$. Then*

1. *if $K \subseteq K'$, then for all $\mathbf{x} \in \mathbb{R}^n$, $\|\mathbf{x}\|_K \geq \|\mathbf{x}\|_{K'}$,*
2. *if $s > 0$, then for all $\mathbf{x} \in \mathbb{R}^n$, $\|\mathbf{x}\|_{sK} = \frac{1}{s}\|\mathbf{x}\|_K$, and*
3. $\|K'\|_K \in \left[\frac{R_{K'}^{(p)}}{R_K^{(p)}}, \frac{R_{K'}^{(p)}}{r_K^{(p)}}\right]$.

We will require the following standard inequality in our work.

Proposition 1 (Brunn-Minkowski)**.** *Let A, B be non-empty compact subsets of $\mathbb{R}^m$. Then $\sqrt[m]{\mathsf{vol}(A+B)} \geq \sqrt[m]{\mathsf{vol}(A)} + \sqrt[m]{\mathsf{vol}(B)}$.*

3.2.3 Lattice Codes

Applications of lattices often require not only a lattice L, but also an efficient algorithm to map arbitrary vectors $\mathbf{x} \in \mathbb{R}^n$ to a nearby lattice point.

Definition 18. *A* lattice code $(L, \lfloor \cdot \rceil)$ *is a lattice $L \subset \mathbb{R}^n$ together with a rounding algorithm $\lfloor \cdot \rceil : \mathsf{span}_{\mathbb{R}}(L) \to L$ such that $\lfloor \mathbf{0} \rceil = \mathbf{0}$ and $\lfloor \mathbf{x} + \mathbf{v} \rceil = \lfloor \mathbf{x} \rceil + \mathbf{v}$ for all $\mathbf{x} \in \mathsf{span}_{\mathbb{R}}(L)$ and $\mathbf{v} \in L$.*

We will be primarily interested in q-ary lattice codes, i.e., lattice codes $(L, \lfloor \cdot \rceil)$ such that L is a q-ary (but not necessarily integer) lattice. For any q-ary lattice code $L \subset \mathbb{R}^n$, we can take the quotients of L and $\mathbb{R}^n$ modulo the additive subgroup $q\mathbb{Z}^n$, and define the *codebook* $L_q = L/q\mathbb{Z}^n$, and *ambient torus* $\mathbb{R}_q^n = (\mathbb{R}/q\mathbb{Z})^n \equiv \mathbb{R}^n/q\mathbb{Z}^n$. Elements of the codebook L_q are called *codewords*, and can be represented as vectors $L \cap [0,q)^n$ with (not necessarily integer) coordinates in the range $[0,q)$. Given a $\mathbb{Z}$-basis of the lattice $\mathbf{B}$, one can moreover represent these codewords as integer via the encoding function $\mathsf{encode}_L(\mathbf{m}) := \mathbf{B}\mathbf{m} \bmod q$, and decoding function $\mathsf{decode}_L(\mathbf{c}) := \mathbf{B}^{-1} \lfloor \mathbf{c} \rceil_L \bmod q$. The codebook L_q is a subgroup of the ambient torus $\mathbb{R}_q^n$, and the rounding function $\lfloor \cdot \rceil : \mathbb{R}^n \to L$ induces a well-defined map $\mathbb{R}_q^n \to L_q$ from the ambient torus to the codebook. Notice that the codebook L_q is a finite set of size $\left|L_q\right| = \frac{q^n}{\det(L)}$, so codewords can be represented with $\lceil \log_2 \left|L_q\right| \rceil \approx n \log q - \log\det(L)$ bits.

For any lattice code $(L, \lfloor \cdot \rceil_L)$, we define the fundamental decoding region $V_L = \{\mathbf{x} \in \mathbb{R}^n \colon \lfloor \mathbf{x} \rceil_L = \mathbf{0}\}$, i.e., the set of all points that decode to $\mathbf{0}$. When $\lfloor \cdot \rceil$ is the CVP rounding function, V_{CVP_L} is called the *Voronoi cell* of the lattice. The reduction of a point $\mathbf{x} \in \mathbb{R}^n$ modulo a lattice code $(L, \lfloor \cdot \rceil_L)$ is defined as $[\mathbf{x}]_L = \mathbf{x} - \lfloor \mathbf{x} \rceil_L$, so that every point in space can be (uniquely) written as the sum $\mathbf{x} = \lfloor \mathbf{x} \rceil_L + [\mathbf{x}]_L$ of a lattice point $\lfloor \mathbf{x} \rceil_L \in L$ and a rounding error $[\mathbf{x}]_L \in V_L$ in the fundamental decoding region. Notice that the rounding error depends not only on the lattice L but also on the rounding function $\lfloor \cdot \rceil$ of the lattice code.

Throughout, we will assume that V_L is a convex symmetric set. When the choice of $\lfloor \cdot \rceil$ is unambiguous, we will refer to the norm $\|\cdot\|_L := \|\cdot\|_{V_L}$, packing radius $r_L^{(p)} := r_{V_L}^{(p)}$, and covering radius $R_L^{(p)} := R_{V_L}^{(p)}$ of L. Note that when $\lfloor \cdot \rceil$ solves CVP on L, the parameters r_L and R_L are the familiar lattice parameters $\lambda_1(L)/2$ and $\rho(L)$. When discussing bounds on the packing/covering radii, we will find it useful to work with normalized (to be invariant to scaling $L \mapsto cL$) versions of these quantities $\overline{r} = (\det L)^{-1/n} r$ and $\overline{R} = (\det L)^{-1/n} R$.

Some Explicit Lattice Codes

We briefly summarize some explicit lattice codes we will use in our work, namely the lattice codes (implicitly) used in previous high-rate constructions of LWE-based encryption [9, 33] (we justify this claim in Section 3.4).

Definition 19 (Primal Gadget Lattice)**.** *For $p, q \in \mathbb{N}$, let $\mathbf{g}_p = (1, p, p^2, \ldots, p^{\lceil \log_p q \rceil - 1})$ be the base-p "gadget vector". The* primal gadget lattice *is the lattice $\Lambda_q(\mathbf{g}_p^t)$.*

Proposition 2. *Let $q = p^\ell$. Then the fundamental region when decoding with Babai's nearest planes $V^{\mathsf{babai}}_{\Lambda_q^\perp(\mathbf{g}_p^t)} = \frac{q}{2p} \cdot \mathscr{B}_\ell^{(\infty)}$, and $\det \Lambda_q(\mathbf{g}_p^t) = q^{\ell-1}$. Moreover, $\det \Lambda_q(\mathbf{g}_p^t) \otimes \mathbb{Z}^{m/\ell} = \det((q/p)\mathbb{Z}^m)$.*

Proof. The fundamental region statement is from [57, Section 4], and the determinant calculation is straightforward.

□

Definition 20 (Dual Gadget Lattice)**.** *For $p, q \in \mathbb{N}$, let $\mathbf{g}_p = (1, p, p^2, \ldots, p^{\lceil \log_p q \rceil - 1})$ be the base-p "gadget vector". The* dual gadget lattice *is the lattice $\Lambda_q^\perp(\mathbf{g}_p^t)$.*

Proposition 3. *Let $p < q$, and let $\ell = \lceil \log_p q \rceil$. Then there exists a decoding algorithms for $\Lambda_q^\perp(\mathbf{g}_p^t)$ that satisfy*

- *when $q = p^\ell$, $r_K^{(\infty)} \geq p/2$,*
- *when $q = p^\ell - 1$, $r_K^{(\infty)} \geq (p-1)/2$,*
- *when $q \in \mathbb{N}$, $r_K^{(\infty)} \geq \frac{(p-1)}{2} \frac{q}{p^\ell}$.*

Proof. The case of $q = p^\ell$ follows from [57]. The case of $q = p^\ell - 1$ follows from [33] (we show that their "nearly square gadget matrix" is the dual gadget lattice in Section 3.4.2). The case of arbitrary q is implicit in [32] (it follows from standard analysis of a decoding algorithm they suggest). We provide this standard analysis below.

Let $S_q = [\mathbf{b}_0, \ldots, \mathbf{b}_{\ell-2}, \mathbf{q}]$ be the typical basis of $\Lambda_q^\perp(\mathbf{g}_p^t)$, where $\mathbf{b}_i = p\mathbf{e}_i - \mathbf{e}_{i+1}$, and $\mathbf{q} = (q_0, \ldots, q_{\ell-1})$ are the base-p digits of q. We abuse notation and state that $q = p^\ell$ has base-p decomposition of $(0, 0, \ldots, 0, p)$. The authors of [32] note that S_q admits a factorization as $S_q = S_{p^\ell} D_q$ where $D_q = [\mathbf{e}_0, \ldots, \mathbf{e}_{\ell-2}, \mathbf{d}_{p,q}]$ for the vector $\mathbf{d}_{p,q}$ with coefficients $\langle \mathbf{e}_i, \mathbf{d}_{p,q} \rangle = \frac{q \bmod p^{i+1}}{p^{i+1}}$. They then suggest using the decoder

$$\mathsf{decode}(x) = S_{p^\ell} \mathsf{decode}_{D_q \mathbb{Z}^\ell}(S_{p^\ell}^{-1} x), \tag{3.3}$$

where one decodes $D_q\mathbb{Z}^\ell$ using Babai's Nearest Planes. This has fundamental region that contains $\frac{q}{p^\ell}[-1/2, 1/2)^\ell$, and therefore the decoder of Eq. (3.3) has fundamental region that contains $S_{p^\ell} \frac{q}{p^\ell}[-1/2, 1/2)^\ell$. One can readily compute that this set contains $(p-1)\frac{q}{p^\ell}[-1/2, 1/2)^\ell$.

We omit the computation of the determinant, as it is straightforward.

□

The next lattice belongs to parameterized family of lattices (for $\mathbf{u}_m^t = (1, 1, \ldots, 1) \in \mathbb{R}^m$) $\Lambda_q(\mathbf{u}_m^t)$ that we call the *Dual of Davenport's Lattice*. Well-known special cases are

- $q = 1$, where it is simply $\mathbb{Z}^m$, and
- $q = 2$, where it is a scaling of D_m^*, the dual of the standard $D_m = \Lambda_2(\mathbf{u}_m^t)$ root lattice.

The generalization to $m > 2$ has been implicit in many works, namely constructing explicit efficient coverings of $\mathbb{R}^m$ [23, Chapter 2, Section 1.3][25], constructing efficient decoding algorithms for certain lattices [29], and constructing rate $1 - o(1)$ fully homomorphic encryption [9].

Definition 21 (Scaled Dual of Davenport's Lattice)**.** *Let $m, q \in \mathbb{N}$. The scaled dual of Davenport's lattice $\Lambda_q(\mathbf{u}_m^t)$ is the lattice $\Lambda_q(\mathbf{u}_m^t) = q\mathbb{Z}^m + \mathbb{Z} \cdot \mathbf{u}_m$, where $\mathbf{u}_m$ is the all-ones vector of length m.*

Definition 22 (A^*_{m-1} Lattice)**.** *For any $m \in \mathbb{N}$, the A^*_{m-1} lattice is defined to be the projection of $\mathbb{Z}^m$ perpendicular to the vector $\mathbf{u}_m$.*

When $m \mid q$, this lattice admits a simple orthogonal decomposition in terms of the root lattice A^*_{m-1}, which admits an $O(m)$-arithmetic operation CVP algorithm [56].

Lemma 25. *Provided $m \mid q$, $\Lambda_q(\mathbf{u}^t_m) = qA^*_{m-1} + \mathbb{Z} \cdot \mathbf{u}_m$, where $\langle qA^*_{m-1}, \mathbb{Z} \cdot \mathbf{u}_m \rangle = \{0\}$.*

Proof. As A^*_{m-1} is defined to be a projection orthogonal to $\mathbf{u}_m$, the last condition is immediate. One can check that $qA^*_{m-1}, \mathbf{u}_m$ are the projections of $\Lambda_q(\mathbf{u}^t_m)$ onto the subspaces perpendicular to and parallel to $\mathbf{u}_m$, respectively, so $qA^*_{m-1} + \mathbf{u}_m \supseteq \Lambda_q(\mathbf{u}^t_m)$. For the other direction, note that $\mathbf{u}_m \subseteq \Lambda_q(\mathbf{u}^t_m)$, as $\Lambda_q(\mathbf{u}^t_m) = q\mathbb{Z}^n + \mathbb{Z} \cdot \mathbf{u}_m$. The equality then immediately follows by [55, Proposition 1.1.6], which implies that the indicies of

- the intersection of $\Lambda_q(\mathbf{u}^t_m) \cap \mathbb{R} \cdot \mathbf{u}_m$ within the projection of $\Lambda_q(\mathbf{u}^t_m)$ onto $\mathbb{R} \cdot \mathbf{u}_m$, and
- the index of $\Lambda_q(\mathbf{u}^t_m)$ within $qA^*_{m-1} + \mathbb{Z} \cdot \mathbf{u}_m$,

are equal. As it is clear that the first index is 1, we have that $\Lambda_q(\mathbf{u}^t_m) = qA^*_{m-1} + \mathbb{Z} \cdot \mathbf{u}_m$.

□

This same argument works for $m \nmid q$, though the indices mentioned in the proof are not all equal to 1. It is fairly straightforward to verify that they are instead equal to $\frac{m}{\gcd(q,m)}$, so one gets an $O(m^2)$-arithmetic operation CVP algorithm for $\Lambda_q(\mathbf{u}^t_m)$ in general. This parameter setting does not appear to be useful for our setting though, as it is unclear how to get any useful information about the shape of the Voronoi cell of the lattice in general.

Proposition 4. *Let $m \mid q$. Then $R^{(\infty)}_{\Lambda_q(\mathbf{u}^t_m)} = \frac{q}{2}\left(1 - \frac{1}{m}\right) + \frac{1}{2}$, and $\det \Lambda_q(\mathbf{u}^t_m) = q^{m-1}$.*

Proof. The orthogonal decomposition implies that $V_{\Lambda_q(\mathbf{u}^t_m)} = V_{qA^*_{m-1}} + V_{\mathbb{Z} \cdot \mathbf{u}_m}$. Applying triangle inequality, we can reduce computing $R^{(\infty)}_{\Lambda_q(\mathbf{u}^t_m)}$ to computing both $R^{(\infty)}_{qA^*_{m-1}}$ and $R^{(\infty)}_{\mathbb{Z} \cdot \mathbf{u}_m}$. The first is straightforward to compute given the explicit expression (found in [23, Chapter 4, Section

6.6]) for $V_{A^*_{m-1}}$, namely as the convex hull of all coordinate permutations of the explicit vector $\mathbf{v} = \frac{1}{2m}(-m+1, -m+3, \ldots, m-3, m-1)$. The second is straightforward to compute as $1/2$.

Finally, to compute the determinant, note that the lattice may be generated by the $m+1$ vectors $[\mathbf{u}_m, q\mathbf{e}_1, \ldots, q\mathbf{e}_m]$, and that any single vector $q\mathbf{e}_i$ can easily be written as a linear combination of the other vectors in this generating set. It follows that $[q\mathbf{e}_1, q\mathbf{e}_2, \ldots, q\mathbf{e}_{m-1}, \mathbf{u}_m]$ is a triangular basis, and $\det \Lambda_q(\mathbf{u}^t_m) = q^{m-1}$.

□

Corollary 2. *If $m \mid q$, one can solve CVP on $\Lambda_q(\mathbf{u}^t_m)$ in $O(m \log_2 q)$ time.*

Proof. Project parallel/perpendicular to $\mathbf{u}_n$, then use the known $O(m)$-arithmetic operation CVP algorithms on qA^*_{m-1} [56] and $\mathbb{Z} \cdot \mathbf{u}_m$. There is an additional $O(\log_2 q)$ overhead as the algorithm of [56] costs arithmetic operations at unit cost.

□

3.2.4 Bounds on Lattice Parameters

For any lattice L, the best normalized packing and covering radii are achieved by the CVP rounding algorithm, giving $\overline{r}_L$ and $\overline{R}_L$. For any m, let $\overline{r}_m = \sup_L \overline{r}_L$ and $\overline{R}_m = \inf_L \overline{R}_L$ be the optimal normalized radii over all lattices L of rank m. It is known that $\overline{r}_m = \Theta(\sqrt{m})$, and $\overline{R}_m = \Theta(\sqrt{m})$ (see Chapters 1 and 2 of [23]). It is additionally known that in each dimension m, there are lattices $L \subseteq \mathbb{R}^m$ that (nearly) simultaneously achieve these bounds, meaning such that $\overline{R}_L / \overline{r}_L \leq 2 + o(1)$, see [12].

3.2.5 Log-Concave Random Variables

We will require the class of *log-concave* random variables.

Definition 23. *Let X be a random variable with pdf $p(x)$. We say that X is* log-concave *if $p(x) = \exp(-V(x))$ for $V(x)$ a convex function.*

We briefly summarize (from [80]) the properties this class of random variables satisfies.

Proposition 5. *Let $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^n$ be log-concave and independent. Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ be any linear transformation. Then $\mathbf{x} + \mathbf{x}'$ and $\mathbf{A}\mathbf{x}$ are log-concave.*

Standard examples of log-concave random variables are Gaussians, and uniform random variables on convex sets K. We establish log concavity of a few other distributions relevant to lattice-based cryptography at the end of this sub-section.

Log-concave random variables are known to have strong concentration properties (they are "sub-exponential"). We use the following concentration bound mostly for simplicity of exposition — one can obtain tighter bounds by treating the cases of the $\|\cdot\|_2$ and $\|\cdot\|_\infty$ norms separately, though as we mention later (Section 3.4) this never impacts our (asymptotic) results.

Proposition 6 (Theorem 11 of [50])**.** *For any L-Lipschitz function $g \in \mathbb{R}^n$, if $\mathbf{x}$ is an isotropic log-concave random variable, then* $\Pr[|g(\mathbf{x}) - \mathbb{E}[g(\mathbf{x})]| > Lt] \leq \exp(-\Omega(t\psi_n^{-1}))$.

Here, ψ_n is *KLS constant*, which is (under the celebrated *KLS conjecture*) $O(1)$ as $n \to \infty$. The current best bound known is $\psi_n = O(\sqrt{\log n})$ [46]. In the rest of our work we will write $\exp(-\tilde{\Omega}(t))$, where this is understood to mean $\exp(-\Omega(t/\sqrt{\log n}))$.

Corollary 3. *If $\mathbf{x}$ is a log-concave random variable in $\mathbb{R}^n$ with covariance matrix Σ, then for $p \in \{2, \infty\}$*

$$\Pr[\|\mathbf{x}\|_p > \sqrt{\mathsf{Tr}(\Sigma)}(t + \sqrt{n})] \leq \exp(-\tilde{\Omega}(t)). \tag{3.4}$$

Proof. Note that $\Sigma^{-1/2}\mathbf{x}$ is isotropic, so we will apply the previous proposition to this random variable and $g(\mathbf{x}) = \|\Sigma^{1/2}\mathbf{x}\|_p$. For the ℓ_2 norm, the Lipschitz constant is the ℓ_2 to ℓ_2 operator norm, i.e. the maximum singular value of $\Sigma^{1/2}$, which is at most $\sqrt{\mathsf{Tr}(\Sigma)}$. For the ℓ_∞ norm, the Lipschitz constant is the ℓ_∞-ℓ_2 operator norm, i.e. the maximum ℓ_2 norm of a column of $\Sigma^{1/2}$. Note that each element of the main diagonal of Σ is the (squared) ℓ_2 norm of a column of Σ, so again we get that $\sqrt{\mathsf{Tr}(\Sigma)}$ bounds the Lipschitz constant.

We therefore have reduced to bounding $\mathbb{E}[g(\mathbf{x})]$ in both cases. For the ℓ_2 norm, by Jenson's inequality, we have that $\mathbb{E}[\|\mathbf{x}\|_2]^2 \leq \mathbb{E}[\|\mathbf{x}\|_2^2] = \mathsf{Tr}(\Sigma)$. For the ℓ_∞ norm, we apply the bound $\mathbb{E}[\|\mathbf{x}\|_\infty] \leq \mathbb{E}[\|\mathbf{x}\|_2] \leq \sqrt{\mathsf{Tr}(\Sigma)}$.

□

We next introduce our *anti-concentration* inequality, which (in a general form) holds for arbitrary polynomials in log-concave random variables. For $t \in \mathbb{R}$ we apply it to the degree-2 polynomial $\|\mathbf{x}\|_2^2 - t$.

Proposition 7 (Theorem 8 of [15])**.** *If* $\mathbf{x}$ *is a log-concave random variable on* $\mathbb{R}^n$ *with covariance matrix* Σ*, then for every* $\varepsilon > 0$*,*

$$\Pr[|\|\mathbf{x}\|_2 - t| \leq \varepsilon] \leq O\left(\frac{\varepsilon}{\sqrt{\mathsf{Tr}(\Sigma)}}\right). \tag{3.5}$$

We end the sub-section by establishing log-concavity of some distributions of cryptographic interest.

Lemma 26. *Let* $\mathbf{e}_i \sim \mathcal{N}(0, \sigma_i^2 I_n)$ *for* $i \in \{0,1\}$*. Then the distribution of* $\langle \mathbf{e}_0, \mathbf{e}_1 \rangle$ *is log-concave if* $n \geq 2$*.*

Proof. By [30, Eq. 2.15], one has that $\langle \mathbf{e}_0, \mathbf{e}_1 \rangle = \frac{\sigma_0 \sigma_1}{2}(X' - X'')$ as distributions, where X', X'' are independent $\chi^2_{(n)}$ random variables. One can easily verify (by directly examining the pdf) that a $\chi^2_{(n)}$ random variable is log-concave if $n \geq 2$. By closure of log concavity under independent sums, the claimed result follows.

□

Theorem 10. *Let* $n \geq 8$*, and let* K *be a bounded measurable subset of* $\mathbb{R}^n$*. Let* $\mathbf{x} \sim \mathcal{N}(0, \sigma^2 I_n)$*, and let* $\mathbf{y} \sim K$ *be independent from* $\mathbf{x}$*. Then* $\langle \mathbf{x}, \mathbf{y} \rangle$ *is log-concave.*

Note that by applying orthogonal transformations to both $\mathbf{x}, \mathbf{y}$, this implies log concavity in the more general case of $\mathbf{x} \sim \mathcal{N}(0, \Sigma)$.

Proof. One can verify that univariate $p(x)$ is log-concave if

$$\forall x : p(x)p''(x) \leq (p'(x))^2. \tag{3.6}$$

We will explicitly compute the pdf of $\langle \mathbf{x}, \mathbf{y} \rangle$, and show that it satisfies this inequality. Note that by the law of total probability

$$\begin{aligned}\mu(A) := \Pr[\langle \mathbf{x}, \mathbf{y} \rangle \in A] &= \int_{\mathbb{R}^n} \Pr[\langle \mathbf{x}, \mathbf{y} \rangle \in A \mid \mathbf{y} = \mathbf{z}] \Pr[\mathbf{y} = \mathbf{z}] d\mathbf{z} \\ &= \int_{\mathbb{R}^n} \left(\int_A \sqrt{2\pi\sigma^2 \|\mathbf{z}\|_2^2}^{-1} \exp\left(-\frac{x^2}{2\sigma^2 \|\mathbf{z}\|_2^2} \right) dx \right) \mathsf{vol}(K)^{-1} \chi_K(\mathbf{z}) d\mathbf{z} \\ &= \int_A \left(\mathsf{vol}(K)^{-1} \int_K \sqrt{2\pi\sigma^2 \|\mathbf{z}\|_2^2}^{-1} \exp\left(-\frac{x^2}{2\sigma^2 \|\mathbf{z}\|_2^2} \right) \right) dx,\end{aligned}$$

where in the last step we applied Fubini's theorem and simplified. If we define $f(x, y^2) = \sqrt{2\pi\sigma^2 y^2}^{-1} \exp\left(-\frac{x^2}{2\sigma^2 y^2} \right)$, it follows that the density is $p(x) = \mathsf{vol}(K)^{-1} \int_K f(x, \|\mathbf{z}\|_2^2) d\mathbf{z} = \mathbb{E}_{\mathbf{z} \leftarrow K}[f(x, \|\mathbf{z}\|_2^2)]$.

To compute the derivatives $p'(x), p''(x)$, we need to interchange differentiation and integration a few times, which we do via the (measure-theoretic) Leibniz Integral rule. Before discussing this, we compute that $\partial_x f(x, y^2) = -\frac{x}{\sigma^2 y^2} f(x, y^2)$, $\partial_x^2 f(x, y^2) = \left(\frac{x^2}{\sigma^4 y^4} - \frac{1}{\sigma^2 y^2} \right) f(x, y^2)$, $\partial_x^3 f(x, y^2) = -\left(\frac{x^3}{\sigma^6 y^6} - \frac{3x}{\sigma^4 y^4} \right) f(x, y^2)$. Our applications of the Leibniz integral rule will require all of these functions (as well as $f(x, y^2)$ itself) to be integrable for all x. The largest singularity occurs when $x = 0$, where $\partial_x^3 f = O(y^{-7})$. As switching to spherical coordinates introduces a multiplicative factor y^{n-1}, provided $n \geq 8$ we can switch to spherical coordinates to get an integrand with no singularity, and show convergence. Note that this step is additionally where we require K to be bounded and measurable, as otherwise $\int_{\mathbb{R}^n} f(x, \|\mathbf{z}\|_2^2) d\mathbf{z} = \infty$ for the same reason that $\int_{\mathbb{R}^n} \|\mathbf{z}\|_2^2 d\mathbf{z} = \infty$. As the other preconditions of Leibniz are straightforward to verify, we omit them.

We next note that one can write

$$\mathbb{E}_{\mathbf{x}}[f(\mathbf{x})] \mathbb{E}_{\mathbf{x}}[g(\mathbf{x})] = \mathbb{E}_{\mathbf{x}, \mathbf{y}} \left[\frac{f(\mathbf{x}) g(\mathbf{y}) + f(\mathbf{y}) g(\mathbf{x})}{2} \right], \tag{3.7}$$

where $\mathbf{y}$ is an i.i.d. copy of $\mathbf{x}$. It follows that

$$(p'(x))^2 = \mathbb{E}_{\mathbf{z}, \mathbf{z}'} \left[\frac{x^2}{\sigma^4} \|\mathbf{z}\|_2^{-2} \|\mathbf{z}'\|_2^{-2} f(x, \|\mathbf{z}\|_2^2) f(x, \|\mathbf{z}'\|_2^2) \right], \tag{3.8}$$

and

$$p(x)p''(x) = \mathbb{E}_{\mathbf{z},\mathbf{z}'}\left[\left(\frac{x^2}{\sigma^4}\left(\frac{\|\mathbf{z}\|_2^{-4}+\|\mathbf{z}'\|_2^{-4}}{2}\right)-\frac{1}{\sigma^2}\left(\frac{\|\mathbf{z}\|_2^{-2}+\|\mathbf{z}'\|_2^{-2}}{2}\right)\right)f(x,\|\mathbf{z}\|_2^2)f(x,\|\mathbf{z}'\|_2^2)\right]. \tag{3.9}$$

Therefore establishing the inequality $(p'(x))^2 \geq p''(x)p(x)$ reduces to showing that some explicit integral is non-negative. Note that $f(x,\|\mathbf{z}\|_2^2) \geq 0$ by inspection. We therefore reduce to showing that the integrand

$$\frac{x^2}{\sigma^4}\|\mathbf{z}\|_2^{-2}\|\mathbf{z}'\|_2^{-2} - \left(\frac{x^2}{\sigma^4}\left(\frac{\|\mathbf{z}\|_2^{-4}+\|\mathbf{z}'\|_2^{-4}}{2}\right)-\frac{1}{\sigma^2}\left(\frac{\|\mathbf{z}\|_2^{-2}+\|\mathbf{z}'\|_2^{-2}}{2}\right)\right) \geq 0. \tag{3.10}$$

This itself follows from the bound $x^{-2}y^{-2} \geq \frac{x^{-4}+y^{-4}}{2}$, valid for any positive x,y, which in the more familiar form $\left(\frac{x^{-4}+y^{-4}}{2}\right)^{-1} \leq x^2y^2$ is simply the inequality between the Harmonic and Geometric means, applied to (x^4,y^4). □

3.2.6 Cryptographic Primitives

We will use the standard notion of IND-CPA security, as well as a less standard notion (that is better suited to lattice-based primitives) known as RND-CPA. Our security analysis in this chapter will not be a bit-security analysis[10], so we will use the (simpler) traditional formulation of the advantage of an adversary in a decision game below.

Definition 24 (IND-CPA)**.** *An encryption scheme* $(\mathsf{KGen},\mathsf{Enc},\mathsf{Dec})$ *is said to be* indistinguishable under chosen plaintext attack *if any efficient (probabilistic polynomial-time) adversary A can only achieve at most negligible advantage in the following game* $\mathscr{G}$*, parameterized by a bit* $b \in \{0,1\}$*:*

1. $k \leftarrow \mathsf{KGen}(1^n)$,
2. $b' \leftarrow A^{O_b(\cdot,\cdot)}$*, where* $O_b(m_0,m_1) = \mathsf{Enc}_k(m_b)$.

[10] In Chapter 4, we will mix computational and statistical primitives, so there will be a concrete benefit to giving a (c,s)-bit security analysis. In this chapter, we instead fix one set of cryptographic parameters (Definition 29) across all instantiations of our framework, and investigate how different coding-theoretic choices will impact the coding-theoretic performance (the rate) of our scheme.

The adversary's advantage is defined to be $\mathsf{adv}(A) = \Delta_{\mathsf{SD}}(A(\mathscr{G}_0), A(\mathscr{G}_1))$.

Definition 25 (RND-CPA)**.** *An encryption scheme* $(\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ *is said to be* pseudorandom under chosen plaintext attack *if one may efficiently produce a uniformly distributed sample from* $C = \{\mathsf{Enc}_k(m) \mid k \in \mathsf{supp}(\mathsf{KGen}(1^n)), m \in \mathscr{M}\}$*, and any efficient (probabilistic polynomial-time) adversary A can only achieve at most negligible advantage in the following game* $\mathscr{G}_b$*, parameterized by a bit* $b \in \{0,1\}$*:*

1. $k \leftarrow \mathsf{KGen}(1^n)$*,*
2. $b' \leftarrow A^{O_b(\cdot)}$*, where* $O_b(m)$ *returns either*
   - $b = 0$*: an encryption* $\mathsf{Enc}_k(m)$ *of the message m under the key k, or*
   - $b = 1$*: a uniform sample from C.*

The adversary's advantage is defined to be $\mathsf{adv}(A) = \Delta_{\mathsf{SD}}(A(\mathscr{G}_0), A(\mathscr{G}_1))$.

Note that the distribution in the $b = 1$ case is not dependent on k, m. A straightforward hybrid argument shows that RND-CPA-security implies IND-CPA-security, although the reverse implication does not hold[11]. We use the (standard) correctness notion of [40], specialized to the setting of private-key encryption.

Definition 26 (δ-Correctness)**.** *A private-key encryption scheme* $(\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ *is said to be* δ*-correct if* $\mathbb{E}_{\mathsf{sk} \leftarrow \mathsf{KGen}(1^n)}[\max_{m \in \mathscr{M}}[\Pr[\mathsf{Dec}_{\mathsf{sk}}(c) \neq m \mid c \leftarrow \mathsf{Enc}_{\mathsf{sk}}(m)]]] \leq \delta$.

3.3 The Encryption Framework

We next present and analyze a secret-key encryption framework. This is done for simplicity of presentation, as the main complication of the public-key setting is a more complex (but, by our results of Section 3.2.5, still log-concave) noise distribution.

To prove bounds in some framework, one must first

[11] Take an IND-CPA-secure cryptosystem, and modify encryption to output $\mathsf{Enc}_k(m) || H(k)$ for a hash function $H(\cdot)$, modeled as a random oracle. As k is not consistent between queries to $O_1(\cdot)$, there is a simple RND-CPA distinguisher, but the construction is still IND-CPA-secure.

$\mathsf{KGen}(1^n)$	$\mathsf{Enc}_{\mathbf{s}}(\mathbf{m})$	$\mathsf{Dec}_{\mathbf{s}}(\mathbf{A},\mathbf{c})$
$\mathbf{s} \leftarrow \chi_{\mathsf{sk}}^n$	$\mathbf{A} \leftarrow \mathbb{Z}_q^{m\times n}$	**return** $\mathsf{decode}_E(\mathbf{c}-\mathbf{As})$
return s	$\mathbf{e} \leftarrow \chi_e^m$	
	$\mathbf{b} = \mathbf{As}+\mathbf{e}+\mathsf{encode}_E(\mathbf{m})$	
	return $(\mathbf{A}, \lfloor \mathbf{b} \rceil_Q)$	

Figure 3.2. Quantized Encryption $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$, where $(E, \lfloor\cdot\rceil_E), (Q, \lfloor\cdot\rceil_Q)$ are lattice codes.

- define a sensible *rate* for the framework, and
- define a *ciphertext error distribution* for the framework.

We do this for our secret-key framework in this section. We additionally show cryptographic security of constructions in our framework, although this is relatively straightforward.

Definition 27 (Quantized LWE Encryption)**.** *Let $(E, \lfloor\cdot\rceil_E), (Q, \lfloor\cdot\rceil_Q)$ be lattice codes in $\mathbb{R}_q^m$. Let χ_{sk} be a distribution on $\mathbb{Z}_q$, and let χ_e be a distribution on $\mathbb{R}_q$. The Quantized LWE Encryption Scheme $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ is given by $(\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$, as defined in Figure 3.2.*

Definition 28. *Let $(E, \lfloor\cdot\rceil_E), (Q, \lfloor\cdot\rceil_Q)$ be lattice codes in $\mathbb{R}_q^m$. Let χ_{sk} be a distribution on $\mathbb{Z}_q$, and χ_e be a distribution on $\mathbb{R}_q$. We say the* asymptotic rate *of $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ is the quantity*

$$\frac{\log_2 |E/q\mathbb{Z}^m|}{\log_2 |Q/q\mathbb{Z}^m|} = 1 - \frac{\log_2 \frac{\det E}{\det Q}}{\log_2 \frac{q^m}{\det Q}}. \tag{3.11}$$

This expression for rate does not include the cost of transmitting $\mathbf{A}$, as there are many ways to reduce (or amortize) this cost, such as appealing to algebraically structured forms of $\mathsf{LWE}_{\chi_{\mathsf{sk}},\chi_e}$, amortizing the cost of $\mathbf{A}$ across many (independent) communication sessions, or transmitting a short seed $s \in \{0,1\}^n$, which one deterministically expands with an extendable output function. In settings where these optimizations are not available (say if one wants to incorporate the cost of transmission of an LWE public key that will be used a single time), one should of course modify the rate to match the particular setting of interest.

We next define the *ciphertext error distribution* of $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$. This is the distribution that E must correct for decryption to succeed.

Lemma 27. *Let $(E,\lfloor\cdot\rceil_E),(Q,\lfloor\cdot\rceil_Q)$ be lattice codes in $\mathbb{R}^m_q$. Let χ_{sk} be a distribution on $\mathbb{Z}_q$, and χ_e be a distribution on $\mathbb{R}_q$. Let $\mathbf{A}\leftarrow\mathbb{Z}_q^{m\times n}$, $\mathbf{e}\leftarrow\chi_e^m$, $\mathbf{s}\leftarrow\chi_{\mathsf{sk}}^n$, and $\mathbf{b}=\mathbf{A}\mathbf{s}+\mathsf{encode}_E(\mathbf{m})+\mathbf{e}$. Then*

$$\mathsf{Dec}_{\mathbf{s}}(\mathsf{Enc}_{\mathbf{s}}(\mathbf{m}))=\mathbf{m}\iff\mathbf{e}-[\mathbf{b}]_Q\in V_E. \tag{3.12}$$

Proof. We have that

$$\begin{aligned}\mathsf{Dec}_{\mathbf{s}}(\mathsf{Enc}_{\mathbf{s}}(\mathbf{m}))&=\mathsf{decode}_E(\lfloor\mathbf{b}\rceil_Q-\mathbf{A}\mathbf{s})\\&=\mathsf{decode}_E(\mathbf{b}-[\mathbf{b}]_Q-\mathbf{A}\mathbf{s})\\&=\mathbf{m}+\mathsf{decode}_E(\mathbf{e}-[\mathbf{b}]_Q).\end{aligned}$$

□

In principle the ciphertext error distribution may depend on $\mathbf{m}$. This and other annoyances (namely that $[\mathbf{b}]_Q$ and $\mathbf{e}$ may be dependent) lead us to introduce the following heuristic description of the ciphertext error distribution.

Heuristic 1. *Let $(Q,\lfloor\cdot\rceil)$ be a lattice code, $\mathbf{m}$ be any message, $\mathbf{c}\in V_Q$, $\mathbf{s}\leftarrow\chi_{\mathsf{sk}}^n$, $\mathbf{A}\leftarrow\mathbb{Z}_q^{m\times n}$, and $\mathbf{e}\leftarrow\chi_e^m$. Then the error $\mathbf{e}-[\mathbf{A}\mathbf{s}+\mathbf{e}+\mathsf{encode}_E(\mathbf{m})]$ is distributed as $\mathbf{e}-\mathbf{u}$, where $\mathbf{u}\leftarrow V_Q$ is independent from $\mathbf{e}$.*

We present a modification of our cryptosystem in Section 3.3.2 that has the same rate as $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$, which (provably) has the above ciphertext error distribution.

We next derive a bound on δ in terms of the scheme parameters. Curiously, we get a better bound if we first separate-off the (bounded) quantization error and apply a *worst-case* bound over this quantity, rather than naively applying Corollary 3.

Lemma 28. *Let $(E, \lfloor\cdot\rceil_E), (Q, \lfloor\cdot\rceil_Q)$ be lattice codes in $\mathbb{R}_q^m$. Let χ_{sk} be a distribution on $\mathbb{Z}_q$, and χ_e be a distribution on $\mathbb{R}_q$. If $\Sigma_{\mathbf{e}}$ is the covariance matrix of $\mathbf{e} \leftarrow \chi_e^m$, $\mathbf{u} \leftarrow V_Q$, then if for some $p \in \{2, \infty\}$, $r_E^{(p)} > \sqrt{\mathsf{Tr}(\Sigma_{\mathbf{e}})}\left(\tilde{O}(\ln(1/\delta)) + \sqrt{m}\right) + R_Q^{(p)}$, it follows that $\mathsf{LWE}_{\chi_{\mathsf{sk}},\chi_e}^{n,q}[E,Q]$ is δ-correct.*

Proof. We have that $\delta = \Pr[\|\mathbf{e}^t - \mathbf{u}^t\|_E > 1] \leq \Pr[\|\mathbf{e}^t\|_E > 1 - \|\mathbf{u}^t\|_E]$. By definition we have that $r_E^{(p)} \cdot \mathscr{B}_m^{(p)} \subseteq V_E$, and therefore for any $\mathbf{x}$, $\|\mathbf{x}\|_E \leq \frac{1}{r_E^{(p)}}\|\mathbf{x}\|_p$. It follows that $\delta \leq \Pr[\|\mathbf{e}^t\|_p > r_E^{(p)} - R_Q^{(p)}]$. Under the assumed bound on $r_E^{(p)}$, our claim follows by Corollary 3.

□

3.3.1 Cryptographic Properties of $\mathsf{LWE}_{\chi_{\mathsf{sk}},\chi_e}^{n,q}[E,Q]$

We next establish RND-CPA security under the $\mathsf{LWE}_{\chi_{\mathsf{sk}},\chi_e}$ assumption. Note that we require no assumptions[12] on E, Q.

Theorem 11. $\mathsf{LWE}_{\chi_{\mathsf{sk}},\chi_e}^{n,q}[E,Q]$ *is* RND-CPA*-secure under the* $\mathsf{LWE}_{\chi_{\mathsf{sk}},\chi_e}$ *assumption.*

Proof. Given an adversary that breaks RND-CPA-security of $\mathsf{LWE}_{\chi_{\mathsf{sk}},\chi_e}$, we describe how to break the decisional $\mathsf{LWE}_{\chi_{\mathsf{sk}},\chi_e}$ assumption. Let $O_b(\cdot)$ be an oracle that either returns samples from (when $b = 0$) $(\mathbf{A}, \mathbf{A}\mathbf{s} + \mathbf{e})$, or (when $b = 1$) $(\mathbf{A}, \mathbf{u}) \leftarrow \mathbb{Z}_q^{n \times m} \times \mathbb{R}_q^m$. Construct an encryption oracle that encrypts $\mathbf{m}$ by

- sampling $(\mathbf{A}, \mathbf{b}) \leftarrow O_b(\cdot)$, and
- returning $(\mathbf{A}, \lfloor \mathbf{b} + \mathsf{encode}_E(\mathbf{m}) \rceil_Q)$.

When $b = 0$, this is exactly the oracle $O_0(\mathbf{m})$ of the RND-CPA game. When $b = 1$, we will show that it is a random ciphertext. Note that $\mathbf{v} := \mathbf{u} + \mathsf{encode}_E(\mathbf{m})$ is the sum of a uniformly random element $\mathbf{u}$ of a group $\mathbb{R}_q^m$ along with an independent element of that group. By a standard argument analogous to the security of the one-time pad, $\mathbf{v}$ is itself uniform over $\mathbb{R}_q^m$, and independent of $\mathsf{encode}_E(\mathbf{m})$. Finally, for uniform $\mathbf{v}$, it is straightforward to see (as $q\mathbb{Z}^m \subseteq Q$) that $\lfloor \mathbf{v} \rceil_Q$ is uniform, finishing the proof.

[12] Part of this claim is an artifact of us using LWE samples with pseudorandom component $\mathbf{b} \in \mathbb{R}_q^m$. If we replace this with $\mathbb{Z}_q^m$, one can establish security if either $E_q \subseteq \mathbb{Z}_q^m$ or $Q_q \subseteq \mathbb{Z}_q^m$. This is still a relatively minor assumption, as it still implies security for E, Q sharing no common (nested) structure.

□

We briefly remark that one could also achieve security of our cryptosystem using a "LWR-type" assumption, namely that $(\mathbf{A}, \lfloor \mathbf{As} \rceil_Q)$ is pseudorandom. This recovers the LWR assumption when Q is a scaling of $\mathbb{Z}^m$.

3.3.2 Quantized LWE Encryption with a Dither

$\mathsf{KGen}(1^n)$	$\mathsf{Enc}_{\mathbf{s}}(\mathbf{m})$	$\mathsf{Dec}_{\mathbf{s}}(\mathbf{A}, \mathbf{c}, \mathbf{v})$
$\mathbf{s} \leftarrow \chi_{\mathsf{sk}}^n$	$\mathbf{A} \leftarrow \mathbb{Z}_q^{m \times n}$	**return** $\mathsf{decode}_E(\mathbf{c} + \mathbf{v} - \mathbf{As})$
return $\mathbf{s}$	$\mathbf{e} \leftarrow \chi_e^m$	
	$\mathbf{b} = \mathbf{As} + \mathbf{e} + \mathsf{encode}_E(\mathbf{m})$	
	$\mathbf{v} \leftarrow V_Q$	
	return $(\mathbf{A}, \lfloor \mathbf{b} - \mathbf{v} \rceil_Q, \mathbf{v})$	

Figure 3.3. Dithered Quantized Encryption $\mathsf{DithLWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E, Q]$, defined relative to lattice codes $(E, \lfloor \cdot \rceil_E), (Q, \lfloor \cdot \rceil_Q)$. Sampling from V_Q can be done efficiently via sampling $\mathbf{v} \leftarrow [0, q)^m$, and then computing $[\mathbf{v}]_Q$.

We next describe a variant of quantized LWE for which Heuristic 1 holds. This utilizes what is known as the *subtractive dither* in coding theory, see Chapter 4 of [87] for more details. Security of our construction easily follows under the same conditions (and proof) of Theorem 11. We omit reproducing this proof for brevity, and instead show that the analogue of Heuristic 1 holds for $\mathsf{DithLWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E, Q]$.

Lemma 29. *Let $(E, \lfloor \cdot \rceil_E), (Q, \lfloor \cdot \rceil_Q)$ be lattice codes in $\mathbb{R}^m_q$. Then the ciphertext error distribution of $\mathsf{DithLWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E, Q]$ satisfies Heuristic 1.*

Proof. For any message $\mathbf{m}$, we can compute that

$$\begin{aligned}
\mathsf{Dec}_{\mathbf{s}}(\mathsf{Enc}_{\mathbf{s}}(\mathbf{m})) &= \mathsf{decode}_E(\lfloor \mathbf{b} - \mathbf{v} \rceil_Q + \mathbf{v} - \mathbf{As}) \\
&= \mathsf{decode}_E(\mathbf{b} - \mathbf{v} - [\mathbf{b} - \mathbf{v}]_Q + \mathbf{v} - \mathbf{As}) \\
&= \mathbf{m} + \mathsf{decode}_E(\mathbf{e} - [\mathbf{b} - \mathbf{v}]_Q).
\end{aligned}$$

Now, as $\mathbf{v}$ is uniform over V_Q, we have that $[\mathbf{b}-\mathbf{v}]_Q$ is uniform over V_Q as well, and independent of $\mathbf{b}$ (and therefore $\mathbf{e}$). It follows that $\mathsf{Dec}_{\mathbf{s}}(\mathsf{Enc}_{\mathbf{s}}(\mathbf{m})) = \mathbf{m}$, unless $\mathbf{e}-\mathbf{u} \notin V_E$, for an independent uniform random variable $\mathbf{u} = [\mathbf{b}-\mathbf{v}]_Q$.

□

We next argue that in practice, $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ and $\mathsf{DithLWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ have the same rate. Recall that we do not explicitly include the random matrix $\mathbf{A} \in \mathbb{Z}_q^{m\times n}$ in our computations of rate. One justification for this was that typically, $\mathbf{A}$ itself is not transmitted, and instead a short seed $s \in \{0,1\}^n$ is transmitted, which is then expanded into $\mathbf{A} := H(s)$ using an extendable output function $H(\cdot)$. If this (common) optimization is used, one can simply generate $\mathbf{v}$ in this same manner, so $\mathbf{v}$ does not need to be explicitly included in ciphertexts.

3.4 Constructions of Quantized LWE Encryption

We next describe the rate achievable by several instantiations (parameterized by lattice codes E,Q) of our framework. The following choice of parameters will be used to enable uniform rate comparisons.

Definition 29. *We say the* standard choice of parameters *are the choice of* $\delta = \exp(-n)$, $\sigma = 2\sqrt{n}$, *and* $m = O(n)$.

3.4.1 Quantizing Regev's Encryption

We first analyze a quantized variant Regev's initial cryptosystem [76] in our framework, namely $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[(q/p)\mathbb{Z}^m, k\mathbb{Z}^m]$ for $k \in \mathbb{N}$. Regev's initial scheme corresponds to the cryptosystem with no quantization ($k = 1$). We will later optimize over the choice of k to attain a rate $1-o(1)$ cryptosystem from polynomial modulus.

Definition 30 (Regev Encryption)**.** *Let* $p,q,k \in \mathbb{N}$. *Regev Encryption is the Quantized LWE encryption scheme* $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[(q/p)\mathbb{Z}^m, k\mathbb{Z}^m]$.

Corollary 4. *Let $p < q$, and $k \in \mathbb{N}$. Then for any $\delta > 0$, provided $\frac{q}{2p} > \tilde{\Omega}(n^{3/2}\sqrt{n+k^2})$, one can parameterize Regev encryption to be δ-correct under the standard choice of parameters and of asymptotic rate at least*

$$1 - O\left(\frac{\log_2(n^2/k)}{\log_2(q/k)}\right). \tag{3.13}$$

We highlight three main takeaways from this example, namely that

1. for trivial quantization ($k = 1$), it is asymptotic rate $1 - \Theta(1)$, i.e. asymptotic rate $1 - \Omega(1)$ from polynomial modulus,

2. for non-trivial quantization ($k = \Omega(n^2)$), it is asymptotic rate $1 - o(1)$ from polynomial modulus, and

3. no parameterization (with polynomially-large q) can achieve asymptotic rate better than $1 - o\left(\frac{1}{\log_2 n}\right)$.

Proof. We get by Lemma 28 that this cryptosystem is δ-correct under the standard choice of parameters provided

$$\frac{q}{2p} > \tilde{\Omega}\left(n^{3/2}\sigma\right) + k. \tag{3.14}$$

Choosing q/p at most a constant-factor larger than this, we get a scheme of asymptotic rate

$$1 - O\left(\frac{\log_2(n^2/k)}{\log_2(q/k)}\right) \tag{3.15}$$

□

We briefly comment on the tightness of our bounds. Prior analysis of ours (not included in this work) that appealed to Gaussian-specific bounds[13] to optimize Eq. (3.14) yielded a different bound on q/p, namely the bound

$$\frac{q}{2p} > \sqrt{2}\sigma(\sqrt{\log_2 m} + \sqrt{\ln n}) + k, \tag{3.16}$$

[13]To handle the (bounded) uniform component $\mathbf{u}$, we appealed to worst-case bounds on its size.

i.e. with no implicit constants[14], and a bound of $q/2p > \Omega(n)$ rather than $q/2p > \tilde{\Omega}(n^2)$. This yields a scheme of asymptotic rate $1 - O\left(\frac{\log_2(n/k)}{\log_2(q/k)}\right)$. We say this to highlight that the more general log-concave analysis (compared to the Gaussian analysis, only relevant for private-key encryption) does result in *some* loss, but only impacts the three points we highlighted above via requiring a larger parameter $k = \Omega(n^2)$.

3.4.2 Quantizing the Cryptosystem of [33]

To demonstrate the breadth of our framework, we next show that it contains the high-rate cryptosystems of [33]. This work proposed two high-rate cryptosystems, namely

- **Section 4.1**: an (unquantized) form of what we call Regev encryption, and
- **Section 4.2**: an (unquantized) form of encryption that uses a lattice generated by a"nearly square gadget matrix" H for error-correction.

As we have already analyzed the first construction, we focus on the second construction in this sub-section. [33] constructs the matrix H as the kernel modulo q of an explicit matrix[15] $F' \otimes I_k$, where (for $q = p^\ell - 1$)

$$F' = \begin{pmatrix} p^{\ell-1} & 1 & \dots & p^{\ell-2} \\ p^{\ell-2} & p^{\ell-1} & & p^{\ell-3} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & p & \dots & p^{\ell-1} \end{pmatrix}. \tag{3.17}$$

One can verify that F' is precisely what one gets when reducing the collection of $\ell+1$ vectors given by $[\mathbf{g}_p, q\mathbf{e}_1, \dots, q\mathbf{e}_\ell]$ to a basis, i.e. is a basis of the lattice $\Lambda_q(\mathbf{g}_p^t)$. It then follows that the desired matrix H is a basis for the lattice $\Lambda_q^\perp(\mathbf{g}_p^t) \otimes \mathbb{Z}^k$ for some k, as we claimed in Table 3.1.

Definition 31 (Gentry-Halevi Encryption, [33])**.** *For $p, q \in \mathbb{N}$, the Gentry-Halevi Encryption scheme is the Quantized LWE encryption scheme* $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[\Lambda_q^\perp(\mathbf{g}_p^t) \otimes \mathbb{Z}^{m/\lceil \log_p q \rceil}, k\mathbb{Z}^m]$.

[14]For this reason, we use this tighter (yet standard) analysis to compute the curves in Figure 3.1.

[15]The matrix we copy down is actually the transpose of the matrix of [33], as we have different conventions for whether lattices are generated by rows/columns of their basis.

Corollary 5. *Let $p < q$, and let $\ell = \lceil \log_p q \rceil$. Assume that $q/p^\ell = O(1)$ with respect to p. Then provided $p > \tilde{\Omega}(n^2) + k$, one can parameterize Gentry-Halevi Encryption to be δ-correct under the standard choice of parameters, and of asymptotic rate at least*

$$1 - O\left(\frac{\log_2(n^2/k)}{\log_2(q/k)}\right). \tag{3.18}$$

Proof. By Lemma 28, this is δ-correct provided $\frac{q}{p^\ell}\frac{(p-1)}{2} > \sqrt{n}\sigma(\tilde{\Omega}(n) + \sqrt{m}) + k$. Under the standard choice of parameters (and assuming $\frac{q}{p^\ell} = O(1)$, independently of p), we get that it suffices to take $p > \tilde{\Omega}(n^2) + k$. This yields a cryptosystem of rate

$$1 - O\left(\frac{\log_2(p/k)}{\log_2(q/k)}\right) = 1 - O\left(\frac{\log_2(n^2/k)}{\log_2(q/k)}\right). \tag{3.19}$$

□

Note that for large-enough $k = \Omega(n^2)$ this is asymptotic rate $1 - o(1)$ from polynomial modulus, while [33] required super-polynomial modulus to attain rate $1 - o(1)$.

3.4.3 Optimizing the Quantized Cryptosystem of [9]

We next consider the only cryptosystem in the literature that uses a quantizer that is not of the form $\mathbb{Z}^{m/k} \otimes Q'$, namely the cryptosystem of [9], which the authors of that work refer to as "linearly homomorphic encryption with ciphertext shrinking". We claim this defines exactly the cryptosystem $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[(q/2)\mathbb{Z}^m, \Lambda_{q/2}(\mathbf{u}_m^t)]$. As this equivalence is not obvious, we briefly recall their construction.

The construction starts with an (unquantized) Regev ciphertext $(\mathbf{A}, \mathbf{As} + \mathbf{e} + (q/2)m)$. It then shows (existentially) that one can find a scalar $r \in \mathbb{Z}_q$ such that the pair $(\mathbf{w} := \mathsf{decode}_{(q/2)\mathbb{Z}^m}(\mathbf{c}_2 + r \cdot \mathbf{u}_m^t), r) \in \mathbb{Z}_2^m \times \mathbb{Z}_q$ suffice for decryption. We view this pair $(\mathbf{w}, r)$ as defining an element of the lattice $\Lambda_{q/2}(\mathbf{u}_m^t) = (q/2)\mathbb{Z}^m + \mathbb{Z} \cdot \mathbf{u}_m$ via the obvious mapping $(\mathbf{w}, r) \mapsto (q/2)\mathbf{w} + r \cdot \mathbf{u}_m$. Note that this mapping is almost a bijection[16]. Under this identification, the pair $(\mathbf{w}, r)$ is simply equal to

[16]When working modulo q, it is instead a bijection between $\mathbb{Z}_2^m \times \mathbb{Z}_{q/2}$ and our lattice, rather than $\mathbb{Z}_2^m \times \mathbb{Z}_q$ and our

$\mathsf{decode}_{\Lambda_{q/2}(\mathbf{u}_m^t)}(\mathbf{c}_2)$ (for a decoding algorithm which need not solve CVP on $\Lambda_{q/2}(\mathbf{u}_m^t)$). If one then attempts to decrypt this ciphertext (using the decryption formula of our work), we have that

$$\begin{aligned}\mathsf{decode}_{(q/2)\mathbb{Z}^m}(\mathsf{encode}_{\Lambda_{q/2}(\mathbf{u}_m^t)}((\mathbf{w},r))-\mathbf{As}) &= \mathsf{decode}_{(q/2)\mathbb{Z}^m}((q/2)\mathbf{w}+r\cdot\mathbf{u}_m-\mathbf{As})\\ &= \mathbf{w}+\mathsf{decode}_{(q/2)\mathbb{Z}^m}(r\cdot\mathbf{u}_m-\mathbf{As})\\ &= \mathsf{decode}_{(q/2)\mathbb{Z}^m}(\mathbf{c}_2+r\cdot\mathbf{u}_m)\\ &\quad-\mathsf{decode}_{(q/2)\mathbb{Z}^m}(\mathbf{As}-r\cdot\mathbf{u}_m).\end{aligned}$$

This is precisely the decryption formula that [9] proposed for their cryptosystem, and therefore their "linearly homomorphic encryption with ciphertext shrinking" is precisely our cryptosystem $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[(q/2)\mathbb{Z}^m,\Lambda_{q/2}(\mathbf{u}_m^t)]$.

We next analyze this construction in our framework, again for a parameterized (by k) family of quantizers that reduces to the cryptosystem of [9] when $k=1$. The family we choose is given by $k\Lambda_{q/(kp)}(\mathbf{u}_m^t)=(q/p)\mathbb{Z}^m+k\mathbf{u}_m^t\cdot\mathbb{Z}$, i.e. we only sparsify the quantizer in a single dimension (parallel to $\mathbf{u}_m^t$). This yields a *much* smaller (non-asymptotic) improvement. We include this more general analysis so we can refer to it during the conclusion.

Our analysis is done where one decodes with respect to the CVP algorithm (Corollary 2) we have previously derived for this lattice.

Definition 32 (Modified BDGM Encryption)**.** *Let $p,q,k\in\mathbb{N}$. The Modified BDGM Cryptosystem is $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[(q/p)\mathbb{Z}^m,k\Lambda_{q/(kp)}(\mathbf{u}_m^t)]$.*

Corollary 6. *For any $\delta>0$, let k be such that $kp\mid q$ and $m\mid q/(kp)$. Then one can parameterize the Modified BDGM Cryptosystem under the standard parameters to be δ-correct, and of asymptotic rate at least*

$$1-O\left(\frac{\log_2(\frac{n^{5/2}}{k})}{m\log_2 p}\right). \tag{3.20}$$

lattice. This extra bit in the r component can be removed from [9], i.e. it is not a difference between our schemes. While saving 1 bit does not matter much, for $p\neq 2$ one will save $\log_2 p$ bits, which can start to matter for $p=\omega(1)$.

Proof. Note that by Proposition 4 we have that $R^{(\infty)}_{k\Lambda_{q/(kp)}(\mathbf{u}_m^t)} \leq \frac{q}{2p}\left(1-\frac{1}{m}\right)+\frac{k}{2}$. By Lemma 28, we have that this cryptosystem is δ-correct provided

$$\frac{q}{2p} > \sqrt{m}\sigma(\tilde{O}(\ln(1/\delta))+\sqrt{m})+\frac{q}{2p}\left(1-\frac{1}{m}\right)+\frac{k}{2}, \tag{3.21}$$

Under standard parameters, this follows provided $q/p \geq \tilde{\Omega}(n^{5/2})+kn$. Choosing q/p that is at most a constant factor larger than this, we get (as $\det k\Lambda_{q/(kp)}(\mathbf{u}_m^t) = k(q/p)^{m-1}$) that the asymptotic rate is at least

$$1-\frac{\log_2 q/kp}{\log_2(q/kp)p^m} \geq 1-\frac{1}{1+m\frac{\log_2 p}{\log_2 q/kp}} \geq 1-O\left(\frac{\log_2(\frac{n^{5/2}}{k})}{m\log_2 p}\right). \tag{3.22}$$

□

We comment the loss in Eq. (3.21) (compared to a Gaussian analysis) is smaller for this scheme — we require $q/p = \tilde{\Omega}(n^{5/2})+kn$ rather than $q/p > \Omega(n^2)+kn$.

3.4.4 Novel Quantized "Gadget" Encryption

We next describe $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[\Lambda_q(\mathbf{g}_p^t)\otimes\mathbb{Z}^{m/\ell},\Lambda_{q/p}(\mathbf{u}_m^t)]$, which combines the quantizer of [9] with the (standard) gadget $\Lambda_q(\mathbf{g}_p^t)\otimes\mathbb{Z}^{m/\ell}$. We find this combination has the exact same rate as [9], while still encoding under an error-correcting code that is a gadget, i.e. we combine the relative strengths of both known constructions of high-rate encryption [33, 9].

Definition 33. *Let $p,q,k\in\mathbb{N}$. The Quantized Gadget Cryptosystem is $\mathsf{LWE}_{\chi_{\mathsf{sk}},\chi_e}[\Lambda_q(\mathbf{g}_p^t)\otimes\mathbb{Z}^{m/\ell},k\Lambda_{q/(kp)}(\mathbf{u}_m^t)]$.*

Corollary 7. *For any $\delta>0$, let k be such that $kp \mid q$ and $m \mid q/(kp)$. Let $q=p^\ell$ for some $\ell>0$. Then one can parameterize the Quantized Gadget cryptosystem under the standard parameters to be δ-correct, and of asymptotic rate at least*

$$1-O\left(\frac{\log_2(\frac{n^{5/2}}{k})}{m\log_2 p}\right). \tag{3.23}$$

Proof. Note that the proof of Corollary 6 only depends on $E=(q/p)\mathbb{Z}^m$ through V_E and $\det E$, and that by Proposition 2 these quantities are equal for $(q/p)\mathbb{Z}^m$ and $\Lambda_q(\mathbf{g}_p^t)\otimes\mathbb{Z}^{m/\ell}$. □

3.5 Rate Impossibility Results

We next establish rate upper bounds (i.e. impossibility) results in two separate noise models, namely that of perfectly correct encryption (with respect to bounded noise), and that of δ-correct encryption (with respect to log-concave noise).

3.5.1 Bounded Noise Model

Recall that (with high probability), a Gaussian $\mathbf{e} \leftarrow \chi_e$ concentrates tightly within a ball of radius $\sigma\sqrt{m}$. We first assume that $\|\mathbf{e}\|_2 \leq \sigma\sqrt{m}$ (say by replacing χ_e^m with a Gaussian that is truncated to be contained in this set), and bound the rate of quantized encryption that has $\delta = 0$, i.e. no decryption failures. This setting is amenable to strong packing arguments.

Theorem 12. *Let $(E, \lfloor\cdot\rceil)$ be a lattice code in $\mathbb{R}^m$. Let χ_e be a distribution such that $\mathsf{supp}(\chi_e^m) = \sqrt{m}\sigma \cdot \mathscr{B}_m$. Then, if $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,\mathbb{Z}^m]$ is 0-correct, it has asymptotic rate at most $1 - \Omega\left(\frac{\log_2(\sqrt{m}\sigma)}{\log_2 q}\right)$, i.e. asymptotic rate $1-o(1)$ encryption from polynomial modulus is impossible.*

Proof. For $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ to be perfectly correct, we need that $\delta = \Pr_{\mathbf{e},\mathbf{b}}[\mathbf{e} - [\mathbf{b}]_Q \notin V_E] = 0$. As we have that $Q = \mathbb{Z}^m$, we have that $[\mathbf{b}]_Q \in [-1/2, 1/2)^m$, and our condition reduces to $\Pr_{\mathbf{e}}[\mathbf{e} + [\mathbf{b}]_Q \notin V_E] = 0$, or equivalently $\Pr_{\mathbf{e}}[\mathbf{e} + [-1/2,1/2)^m \subseteq V_E] = 1$, i.e. $\mathsf{supp}(\chi_e^m) \subseteq \mathsf{supp}(\chi_e^m) + [-1/2,1/2)^m = \sqrt{m}\sigma \cdot \mathscr{B}_m + [-1/2,1/2)^m \subseteq V_E$.

Now, as $E_q + V_E = \mathbb{R}_q^m$ is a partition, we have that $E_q + \sqrt{m}\sigma \cdot \mathscr{B}_m \subseteq \mathbb{R}_q^m$ is a packing, meaning the sets $\{e + \sqrt{m}\sigma \cdot \mathscr{B}_m\}_{e \in E_q}$ are disjoint. Taking volumes of both sides, we have that

$$\mathsf{vol}(E_q + \mathsf{supp}(\chi_e^m)) \overset{1}{=} |E_q| \mathsf{vol}(\sqrt{m}\sigma \cdot \mathscr{B}_m) \leq q^m = \mathsf{vol}(\mathbb{R}_q^m), \tag{3.24}$$

where (1) easily follows from the aforementioned disjointness condition.

Now, we have that $\left|E_q\right| = \frac{q^m}{\det E}$. Rearranging, we get that $\det E \geq \mathsf{vol}(\sqrt{m}\sigma \cdot \mathscr{B}_m)$. Stirling's approximation gives that $\mathsf{vol}(\sqrt{m}\sigma \cdot \mathscr{B}_m) \approx \frac{1}{\sqrt{m\pi}} \left(\frac{2\pi e}{m}\right)^{m/2} (\sqrt{m}\sigma)^m$. Finally, we have that the

asymptotic rate is

$$R = 1 - \frac{\log_2 \frac{\det E}{\det \mathbb{Z}^m}}{\log_2 \frac{q^m}{\det \mathbb{Z}^m}} = 1 - \frac{\log_2 \det E}{m \log_2 q} \leq 1 - \Omega\left(\frac{\log_2(\sqrt{m}\sigma)}{\log_2 q}\right). \tag{3.25}$$

□

Theorem 13. *Let $(E, \lfloor \cdot \rceil_E)$, and $(Q, \lfloor \cdot \rceil_Q)$ be lattice codes in $\mathbb{R}^m$. Let χ_e be a distribution such that $\mathsf{supp}(\chi_e^m) = \sqrt{m}\sigma \cdot \mathscr{B}_m$. Assume that Heuristic 1 holds. Then, if $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ is 0-correct, it has asymptotic rate at most*

$$1 - \frac{\log_2(1 + \frac{\sqrt{2\pi e}\sigma}{\sqrt[m]{\det Q}})}{\log_2 \frac{q}{\sqrt[m]{\det Q}}}. \tag{3.26}$$

Proof. For $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ to be perfectly correct, we need that $\delta = \Pr_{\mathbf{e},\mathbf{b}}[\mathbf{e} - [\mathbf{b}]_Q \notin V_E] = 0$. Equivalently, we need that $\Pr_{\mathbf{e},\mathbf{b}}[\mathbf{e} - [\mathbf{b}]_Q \in V_E] = 1$. Under Heuristic 1, we have that the random variable $\mathbf{e} - [\mathbf{b}]_Q$ has support $\mathsf{supp}(\chi_e^m) + (-V_Q)$. Note that V_Q is centrally symmetric, so $-V_Q = V_Q$. We therefore have that $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ is 0-correct if and only if $\sqrt{m}\sigma \cdot \mathscr{B}_m + V_Q \subseteq V_E$.

Now, as $E_q + V_E = \mathbb{R}_q^m$ is a partition, we have that $E_q + (\sqrt{m}\sigma \cdot \mathscr{B}_m + V_Q) \subseteq \mathbb{R}_q^m$ is a packing, i.e. the sets $\{e + (\sqrt{m}\sigma \cdot \mathscr{B}_m + V_Q)\}_{e \in E_q}$ are disjoint. Taking volumes, we have that

$$\mathsf{vol}(E_q + (\sqrt{m}\sigma \cdot \mathscr{B}_m + V_Q)) = \left|E_q\right| \mathsf{vol}(\sqrt{m}\sigma \cdot \mathscr{B}_m + V_Q) \leq q^m = \mathsf{vol}(\mathbb{R}_q^m). \tag{3.27}$$

As $\left|E_q\right| = \frac{q^m}{\det E}$, this inequality is equivalent to $\sqrt[m]{\det E} \geq \sqrt[m]{\mathsf{vol}(\sqrt{m}\sigma \cdot \mathscr{B}_m + V_Q)}$. Applying the Brunn-Minkowski inequality (Proposition 1) and Stirling's Approximation, we get that

$$\sqrt[m]{\det E} \geq \sqrt{2\pi e}\sigma + \sqrt[m]{\det Q}. \tag{3.28}$$

This immediately implies that the asymptotic rate is

$$R = 1 - \frac{\log_2 \frac{\det E}{\det Q}}{\log_2 \frac{q^m}{\det Q}} = 1 - \frac{\log_2(1 + \frac{\sqrt{2\pi e}\sigma}{\sqrt[m]{\det Q}})}{\log_2 \frac{q}{\sqrt[m]{\det Q}}}. \tag{3.29}$$

□

Note that the upper bound becomes $1 - o\left(\frac{1}{\log_2 \frac{q}{\sigma}}\right)$ if $\sqrt[m]{\det Q} \approx \sigma$, i.e. rate $1 - o(1)$ encryption is no longer impossible provided one quantizes even a relatively small amount.

3.5.2 Results for Unbounded Errors

We next return to the setting of χ_e an arbitrary log-concave distribution, and bounding δ-correct encryption for $\delta > 0$. Here, we rely on the anti-concentration inequality of Proposition 7, rather than the prior packing arguments. We first give a bound that is mostly useful in the case of trivial quantization, i.e. where $Q = \mathbb{Z}^m$.

Theorem 14. *Let $\varepsilon > 0$. Let $(E, \lfloor \cdot \rceil_E)$, $(Q, \lfloor \cdot \rceil_Q)$ be any lattice codes in $\mathbb{R}^m$. Let the ciphertext error distribution has covariance matrix Σ. If $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ is δ-correct, then the asymptotic rate of $\mathsf{LWE}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ is at most*

$$1 - \frac{\log_2 \Omega\left(\frac{\sqrt{\mathsf{Tr}(\Sigma)}}{\bar{R}_E}\right)}{\log_2 q / \sqrt[m]{\det Q}} + o(1). \tag{3.30}$$

Proof. We have that

$$1 - \delta \leq \Pr_{\mathbf{e}}[\|\mathbf{e}\|_E \leq 1] \leq \Pr_{\mathbf{e}}[\|\mathbf{e}\|_2 \leq R_E] \leq O\left(\frac{R_E}{\sqrt{\mathsf{Tr}(\Sigma)}}\right).$$

The first inequality is from Lemma 24, and the second from Proposition 7. We then easily get the bound $\sqrt[m]{\det E} \geq \Omega\left(\frac{1-\delta}{\bar{R}_E}\sqrt{\mathsf{Tr}(\Sigma)}\right)$, and the asymptotic rate is

$$R = 1 - \frac{\log_2 \sqrt[m]{\det E}}{\log_2 q / \sqrt[m]{\det Q}} \leq 1 - \frac{\log_2 \Omega\left(\frac{\sqrt{\mathsf{Tr}(\Sigma)}(1-\delta)}{\bar{R}_E}\right)}{\log_2 q / \sqrt[m]{\det Q}}, \tag{3.31}$$

Finally, we separate off the $1 - \delta$ term, and note that $-\log_2(1-\delta)/\log_2 q$ is easily $o(1)$ to get the claimed result.

□

The presence of $\overline{R}_E$ in this bound is peculiar, and we cannot remove it by appealing to a universal upper bound on $\overline{R}_E$ (no such bound exists, even if we restrict V_E to be the Voronoi cell of a lattice). If we assume $\overline{R}_E$ is not too large (either absolutely, or in comparison to $\overline{r}_E$), we can prove impossibility of rate $1-o(1)$ encryption.

Corollary 8. *Let $\varepsilon > 0$, and let $(E, \lfloor\cdot\rceil_E)$ be a lattice code in $\mathbb{R}^m$. If either*

- $\overline{R}_E \leq O(m^{1-\varepsilon})$, *or*
- $\overline{R}_E/\overline{r}_E \leq O(m^{1/2-\varepsilon})$,

and q is polynomially large, then $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,\mathbb{Z}^m]$ is of rate $1-\Omega(1)$, i.e. under these conditions rate $1-o(1)$ encryption is impossible.

Proof. We show that the second condition implies the first. This is simple, as the bound $\overline{r}_E \leq O(m^{1/2})$ implies that $\overline{R}_E \leq O(\overline{r}_E m^{1/2-\varepsilon}) \leq O(m^{1-\varepsilon})$. Next, note that by Theorem 14, we have that the asymptotic rate is at most

$$1 - \frac{\log_2 \Omega\left(\frac{\sqrt{m}\sigma}{m^{1-\varepsilon}}\right)}{\log_2 q} + o(1) = 1 - \varepsilon\frac{\log_2 \Omega(m)}{\log_2 q} - \frac{\log_2 \frac{\sigma}{\sqrt{m}}}{\log_2 q} + o(1). \tag{3.32}$$

As q is polynomially large, this suffices for the claimed result.

□

Corollary 9. *There exist lattice codes E with $\overline{r}_E \geq \Omega(\sqrt{m})$, i.e. within a constant factor of optimal, such that $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,\mathbb{Z}^m]$ is of rate $1-\Omega(1)$.*

Proof. Choose E with $\frac{\overline{R}_E}{\overline{r}_E} \leq 2 + o(1)$, which are known to exist [12], and then apply Corollary 8.

□

Therefore, any result establishing rate $1-o(1)$ encryption from $Q = \mathbb{Z}^m$ and $q = n^{O(1)}$ must do more than simply appeal to the packing radius $\overline{r}_E = \Theta(\sqrt{m})$ being nearly optimal.

We next extend our bound on $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ for ciphertext error distribution the sum of a log-concave random variable and $\mathbf{u} \leftarrow V_Q$ uniform, in a similar way to how we got sharper upper bounds on δ by considering this special case.

Theorem 15. *Let $(E, \lfloor\cdot\rceil_E), (Q, \lfloor\cdot\rceil_Q)$ be lattice codes in $\mathbb{R}^m$, and assume that $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ is δ-correct. Assume that Heuristic 1 holds, i.e. one can write the ciphertext error distribution as the independent sum of a log-concave random variable (with covariance matrix Σ) and $\mathbf{u} \leftarrow V_Q$. Then $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ is of asymptotic rate at most*

$$1 - \frac{\log_2 \Omega\left(\frac{\sqrt{\mathsf{Tr}(\Sigma)}}{R_Q}\right)}{m \log_2 \frac{q}{\sqrt[m]{\det Q}}} + o(1). \tag{3.33}$$

Proof. Throughout, let $p(x)$ be the density of the log-concave random variable $\mathbf{e}$. By the law of total probability, we have that

$$\begin{aligned}
\Pr[\mathbf{e} - \mathbf{u} \in V_E] &= \frac{1}{\det Q} \int_{V_Q} \int_{V_E} p(\mathbf{e} - \mathbf{x}) d\mathbf{e} d\mathbf{x} \\
&\leq \frac{1}{\det Q} \int_{V_E} \Pr[\|\mathbf{e} - \mathbf{x}\|_2 \leq R_Q] d\mathbf{e} \\
&\leq O\left(\frac{\det E}{\det Q} \frac{R_Q}{\sqrt{\mathsf{Tr}(\Sigma)}}\right),
\end{aligned}$$

where the first inequality is the containment $V_Q \subseteq R_Q \cdot \mathscr{B}_n$ (as well as Fubini's theorem), and the second inequality is Proposition 7. It follows that the asymptotic rate is

$$1 - \frac{\log_2 \Omega\left(\frac{\sqrt{\mathsf{Tr}(\Sigma)}}{R_Q}\right)}{\log_2 |Q/q\mathbb{Z}^m|} - \frac{\log_2(1-\delta)}{\log_2 |Q/q\mathbb{Z}^m|}. \tag{3.34}$$

We finish by applying the same bound to $1 - \delta$ as we did in Theorem 14.

□

Corollary 10. *Let $(E, \lfloor\cdot\rceil_E), (Q, \lfloor\cdot\rceil_Q)$ be lattice codes in $\mathbb{R}^m$, and let $\varepsilon > 0$. Assume the validity of Heuristic 1. If $\overline{R}_Q \leq O(\sqrt{m})$ is within a constant factor of optimal, then the asymptotic rate of $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ is at most*

$$1 - \frac{\log_2 \Omega\left(\frac{\sigma}{\sqrt[m]{\det Q}}\right)}{m \log_2 \frac{q}{\sqrt[m]{\det Q}}} + o(1). \tag{3.35}$$

In particular, if $\sqrt[m]{\det Q} \le O(\sigma)$*, this quantity is at most* $1 - \Omega\left(\frac{1}{m \log_2 \frac{q}{\sigma}}\right) + o(1)$.

Proof. This follows directly from plugging the bounds we assume into Theorem 15.

□

Note that, as our modification of BDGM encryption (Corollary 6) and the Quantized Gadget cryptosystem (Corollary 7) have rate $1 - O\left(\frac{1}{m}\right)$, under the standard choice of parameters this bound is tight up to an $O(\log_2 m)$ factor for quantizers with $\sqrt[m]{\det Q} \le O(\sigma)$.

3.5.3 Exponentially Stronger Bounds Against a Common Design Paradigm

We finish by showing that the bound Corollary 10 can be significantly strengthened when restricting to $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ where $E = \mathbb{Z}^{m/\dim E'} \otimes E'$, $Q = \mathbb{Z}^{m/\dim E'} \otimes Q'$ are the direct sum of m/k identical (smaller) codes for $k = \dim E' = \dim Q'$. In what follows we *solely* change the dimension, and keep the other parameters q, δ, σ, n fixed.

Lemma 30. *Let* $E = \mathbb{Z}^{m/k} \otimes E'$*, and* $Q = \mathbb{Z}^{m/k} \otimes Q'$*, where* E', Q' *are* k*-dimensional lattice codes. Then the asymptotic rate of* $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E,Q]$ *is equal to the asymptotic rate of* $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E',Q']$.

Proof. Note that $\det E = (\det E')^{m/k}$, and similarly for $\det Q$. We then have that the asymptotic rate is

$$\frac{\log_2 \frac{q^m}{\det E}}{\log_2 \frac{q^m}{\det Q}} = \frac{\log_2 \frac{q^m}{(\det E')^{m/k}}}{\log_2 \frac{q^m}{(\det Q')^{m/k}}} = \frac{\log_2 \frac{q^k}{\det E'}}{\log_2 \frac{q^k}{\det Q'}}. \tag{3.36}$$

□

Corollary 11. *Let* $(E', \lfloor\cdot\rceil_{E'}), (Q', \lfloor\cdot\rceil_{Q'})$ *be lattice codes in* $\mathbb{R}^k$*, let* $k \mid m$*, and let* $\varepsilon > 0$*. Assume the validity of Heuristic 1. If* $\overline{R}_{Q'} \le O(\sqrt{k})$ *is within a constant factor of optimal, then the asymptotic rate of* $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[\mathbb{Z}^{m/k} \otimes E', \mathbb{Z}^{m/k} \otimes Q']$ *is at most*

$$1 - \frac{\log_2 \Omega\left(\frac{\sigma}{\sqrt[k]{\det Q}}\right)}{k \log_2 \frac{q}{\sqrt[k]{\det Q}}} + o(1). \tag{3.37}$$

In particular, if $\sqrt[k]{\det Q'} \le O(\sigma)$*, this quantity is at most* $1 - \Omega\left(\frac{1}{k \log_2 \frac{q}{\sigma}}\right) + o(1)$.

Proof. Use Corollary 10 to bound the rate of $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[E',Q']$. By Lemma 30, this implies the same bound for $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[\mathbb{Z}^{m/k}\otimes E',\mathbb{Z}^{m/k}\otimes Q']$.

□

Note that in the literature, k is typically at most $O(\log_2 m)$, so this bound is exponentially stronger than Corollary 10 in this common setting.

3.6 Conclusion and Open Problems

Conclusion

We propose a framework that reduces the design of LWE-based encryption to a handful of coding-theoretic choices. We then prove bounds on any instantiation of this framework, and find that a preexisting cryptosystem in the literature [9] is within an $O(\log_2 m)$ factor of optimal rate. We additionally prove bounds against the common situation of building lattices for error-correction and quantization by setting $L=\bigoplus_{i=1}^{m/\log_2 m} L'$ for $\dim L' = \Theta(\log_2 m)$. We establish exponentially stronger bounds against this setting, which we validate via practical rate computations.

Open Problems

We find an $O(\log_2 m)$ gap between the best-known construction and our bound for any construction. This gap is surprisingly significant — if there exists a construction meeting our bound, it implies constant (independent of the amount of data to transmit) overhead lattice-based encryption, i.e. a lattice-based cryptosystem that is similar to (standard) hybrid encryption. Does such a cryptosystem exist, or can one establish the impossibility of such a construction? Note that our cryptosystem $\mathsf{LWE}^{n,q}_{\chi_{\mathsf{sk}},\chi_e}[(q/p)\mathbb{Z}^m, k\Lambda_{q/(kp)}(\mathbf{u}_m^t)]$ gets quite close. If we did not have the divisibility requirement $m \mid q/(kp)$, it would suffice to close the gap itself. Can this requirement be removed? Finally, our work suggests the quantizer $\Lambda_{q/p}(\mathbf{u}_m^t)$ is much better than $k\mathbb{Z}^m$, which is implicitly used to define the LWR assumption. Can one obtain secure and practical LWR-type constructions using this quantizer?

3.7 Acknowledgments

Chapter 3, in full, is a reprint of the material as it appears in Advances in Cryptology — CRYPTO 2023. Micciancio, Daniele; Schultz-Wu, Mark. "Error Correction and Ciphertext Quantization in Lattice Cryptography". The dissertation author was a primary investigator and author of this material.

Chapter 4

Securing Approximate Homomorphic Encryption using Differential Privacy

4.1 Chapter Introduction

Fully homomorphic encryption (FHE) on *approximate* numbers, proposed by Cheon, Kim, Kim and Song in [19], has attracted much attention in the past few years as a method to improve the efficiency of computing on encrypted data in a wide range of applications (like privacy preserving machine learning) where approximate results are acceptable [21, 18, 17, 22, 16, 38, 65]. The CKKS scheme [20], just like most other (homomorphic) encryption schemes based on lattices, can be proved to satisfy the well established security notion of *indistinguishability under chosen plaintext attack* (IND-CPA) [35] under widely accepted complexity assumptions, like the average-case hardness of the *Learning With Errors (LWE)* problem or the worst-case complexity of computational problems on (algebraic) point lattices [77, 54, 68, 67].

Recently Li and Micciancio [52] have shown that the traditional formulation of IND-CPA security is inadequate to capture security of approximate encryption against passive attacks, and demonstrated that the CKKS scheme is susceptible to a very efficient total key recovery attack, mounted by a passive adversary. The problem highlighted in [52] is not with the IND-CPA security definition per se, which remains a good and well accepted definition for exact FHE schemes, but with the specifics of approximate decryption, which may inadvertently leak information about the secret key even when used by honest parties. The work [52] also proposes a new, enhanced

formulation of IND-CPA security (called IND-CPA$^{\mathsf{D}}$, or IND-CPA *with decryption oracles*), which properly captures the capabilities of a passive attacker against an *approximate* FHE scheme, and is equivalent to the standard notion of IND-CPA security for encryption schemes with exact decryption. The work [52] also suggested some practical countermeasures to avoid their attack, and all major open source libraries implementing CKKS (e.g., [81, 39, 64, 48]) included similar countermeasures shortly after the results in [52] were made public. However, neither [52] nor any of these libraries present a solution that provably achieves the IND-CPA$^{\mathsf{D}}$ security definition proposed in [52], leaving it as an open problem.

4.1.1 Our Results and Techniques

In this work we show how to achieve IND-CPA$^{\mathsf{D}}$ security in a provable way. More specifically, we present a general technique to transform any approximate FHE scheme satisfying the (weak) IND-CPA security notion into one achieving the strong IND-CPA$^{\mathsf{D}}$ security definition proposed in [52]. We then demonstrate how to apply the technique to the specific case of the CKKS scheme, which is the most prominent example of approximate homomorphic encryption.

Our technique works by combining a given (approximate) FHE scheme with another fundamental tool from the cryptographers' toolbox: differential privacy. The construction is very simple and intuitive: given an approximate FHE scheme (like CKKS), we modify the decryption function by post-processing its output (the decrypted message) with a properly chosen differentially private mechanism. Using differential privacy to limit the key leakage of approximate decryption is a fairly natural idea, and it is essentially the intuition behind the practical countermeasures proposed in [52] and implemented by the libraries. But formally analyzing the method and provably achieving IND-CPA$^{\mathsf{D}}$ security raises a number of technical challenges:

- The Hamming metric, commonly used to define and analyze differentially private mechanisms, is not well suited to the setting of (lattice based) homomorphic encryption.

- Similarly, the Laplace noise commonly used and studied in the standard setting of differential privacy is not a good match for our target application, as it is both associated with the wrong

norm (ℓ_1, rather than ℓ_2 or ℓ_∞), and has heavier tails than, e.g., the Gaussian distribution, and so will give worse bounds on the error introduced by post-processing.

- Formally proving the security of our construction requires a careful definition of what it means for an FHE scheme to be *approximate*. Previous works [20, 52] simply defined approximate FHE as an encryption scheme which *does not* satisfy the correctness requirement

$$\mathsf{Dec}(\mathsf{Eval}(f, \mathsf{Enc}(m_1), \ldots, \mathsf{Enc}(m_k))) = f(m_1, \ldots, m_k) \tag{4.1}$$

  without imposing any specific limitation on how a scheme may deviate from it.

- Perturbing the output of the decryption function with a differentially private mechanism comes at the cost of lowering the output quality, making the result of the (already approximate) decryption function even less accurate, highlighting the necessity of carefully tuning the amount of noise added.

- The minimal security level considered acceptable by applications in practice typically depends on whether the cryptographic primitive is statistically secure (against computationally unbounded adversaries) or computationally secure (in which case a higher security margin is advisable to anticipate possible algorithmic or implementation improvements in the attacks). Our application of statistical security tools (differential privacy) to encryption seems to require the instantiation of statistical security with the high security parameters of a computational encryption scheme.

In order to address the above obstacles, we

- provide a general definition of differential privacy, parameterized by an arbitrary norm, and then instantiate it with the Euclidean norm for the case of lattice-based encryption;

- employ a differentially private mechanism (for the Euclidean norm) based on Gaussian noise, which blends well with the probability distributions used in lattice cryptography;

- give formal definitions of *approximate* FHE, which provide precise guarantees on the output quality of the (approximate) decryption function. In fact, we identify two possible definitions, based on what we call *static* and *dynamic* noise estimates, and show that they result in quite different security properties (more on this below);

- use KL-divergence and other probabilistic tools to carefully calibrate the mechanism noise to the output quality, showing that $\Theta(c)$ bits of noise are required to formally achieve c-bit $\mathsf{IND\text{-}CPA}^{\mathsf{D}}$ security;

- leverage the finer grained definition of (c,s)-bit-security[1] that distinguishes between a computational security parameter c and a statistical one s, which can be set to a lower value than c.

We first elaborate on our definition of *approximate* FHE. Previous works [20, 52] did not include a precise definition of what it means for an encryption scheme (or decryption function) to be approximate, because the quality of the approximation (and more generally, the definition of the decryption function itself) does not impact the IND-CPA security of a scheme. This is contrasted with our work, where bounding the approximation quality of the decryption function plays a critical role in our analysis. Generally speaking, an approximate FHE scheme provides a guarantee (upper bound) on how much the output of the decryption function $\mathsf{Dec}(\mathsf{Eval}(f,\mathsf{Enc}(m_1),\ldots,\mathsf{Enc}(m_k)))$ may deviate from the output of the computation $f(m_1,\ldots,m_k)$. We distinguish two types of approximate FHE:

- Approximate FHE with *static* noise estimates, where this bound can be publicly computed as a function of the homomorphic computation f performed on the input ciphertexts. This is, for example, the type of noise estimates used in the HElib library [39].

- Approximate FHE with *dynamic* noise estimates, where this bound is computed by the decryption function Dec using also the input ciphertext and the secret key. An ingenious

[1] While this is defined in Chapter 2 in this document, the definition of (c,s)-bit security initially occurred in [53], whose contents make up this chapter.

method for dynamic noise estimation has been proposed by the PALISADE library [64].

Most of our results, like our general framework based on differential privacy and a provably IND-CPA$^{\mathsf{D}}$ secure variant of the CKKS approximate FHE scheme, are in the setting of static noise estimates. In this setting, we are able to establish the security of our generic construction (Theorem 16), and provide precise security guarantees for the modified approximate FHE scheme, showing that if the original scheme is c-bit IND-CPA secure, then combining it with an appropriate differentially private mechanism achieves $c - \log_2 6$ bits of security against the stronger IND-CPA$^{\mathsf{D}}$ security definition, losing only ≈ 3 bits of security (Theorem 16). The amount of noise required to achieve this result is quantified by the notion of ρ-KLDP (Kullback-Leibler Differential Privacy), for a sufficiently small value of ρ. Our analysis is nearly tight for the CKKS scheme, in the sense that if one uses a substantially smaller amount of noise, we are able to exhibit an attack that breaks IND-CPA$^{\mathsf{D}}$ security (Theorem 18).

When setting the parameters of a cryptosystem (or other computational cryptographic primitive), it is common to use a very conservative security level to anticipate reductions in both the hardware and operational cost of mounting an attack. A common level of security considered adequate for most applications is $c = 128$ bits of security. When applying a statistical technique (like differential privacy) to a computational primitive, this seems to require instantiating the statistical technique with the same (high) level of bit security. We leverage the notion of (c,s)-bit security of Chapter 2, which is parameterized by both a *computational* parameter c and *statistical* parameter s. (c,s)-security is technically easier to achieve than both c-bit computational security, and s-bit statistical security, and allows us to decrease the cost of our countermeasure (Theorem 16) by lowering the required amount of DP noise by $(c-s)/2$ bits. The standard notion of bit-security corresponds to setting $s = c$, which gives no improvement. But for typical parameter settings (e.g., $c = 128$ and $s = 64$), the refined definition allows to reduce the required amount of noise from ≈ 75 bits to ≈ 45, a substantial saving of ≈ 30 bits. As even more conservative choices, such as $s = 80$ or $s = 100$, yield savings of ≈ 24 or ≈ 14 bits of noise, we expect this refined notion of security to be concretely useful when securing CKKS against the attacks of [52].

All this is for static noise estimates. Dynamic estimates are interesting because they can provide stronger (probabilistic) guarantees on the output quality of the decryption function. Interestingly, we show that the same intuitive idea of combining approximate FHE with differential privacy, while calibrating the DP noise via dynamic error estimates, does not result in a secure scheme. In particular, we describe attacks to the $\mathsf{IND\text{-}CPA}^{\mathsf{D}}$ security of CKKS using dynamic noise estimates (Theorem 19), and complete key recovery attacks for other (artificially constructed) IND-CPA-secure FHE schemes (Theorem 20).

4.1.2 Chapter Outline

The rest of the chapter is organized as follows. In Section 4.2 we present background definitions and results from cryptography, fully homomorphic encryption, and probability theory. In Section 4.3 we present our general framework to secure approximate FHE using differential privacy, for the setting of *static* error estimation. In Section 4.4 we apply the framework to the CKKS scheme, and develop our relaxed notion of bit security. In Section 4.5 we present our (negative) results for approximate FHE with *dynamic* error estimation. Section 4.6 concludes with a summary of our results and open problems.

4.2 Chapter Preliminaries

We recall some notions and known results.

4.2.1 Fully Homomorphic Encryption

We briefly review definitions related to FHE. For simplicity, we focus on public-key setting. In all our definitions, we denote the (computational) security parameter using c.

Definition 34 (FHE Scheme)**.** *A (public-key) homomorphic encryption scheme with plaintext space $\mathscr{M}$, ciphertext space $\mathscr{C}$, public key space $\mathscr{PK}$, secret-key space $\mathscr{SK}$, and space of evaluatable*

circuits $\mathscr{L}$ is a tuple of four probabilistic polynomial-time algorithms

$$\mathsf{KeyGen} : 1^{\mathbb{N}} \to \mathscr{PK} \times \mathscr{SK}$$

$$\mathsf{Enc} : \mathscr{PK} \times \mathscr{M} \to \mathscr{C}$$

$$\mathsf{Dec} : \mathscr{SK} \times \mathscr{C} \to \mathscr{M}$$

$$\mathsf{Eval} : \mathscr{PK} \times \mathscr{L} \times \mathscr{C} \to \mathscr{C}$$

Typically the public key naturally splits into two components, one used by Enc and one used by Eval. This separation is used to minimize the storage requirements of encryption (as the evaluation key is often quite large), and has no impact on security, so for simplicity we model both Enc and Eval as taking as input the same public key.

Standard FHE schemes are expected to satisfy the following notion of correctness.

Definition 35 (Perfect Correctness)**.** *An FHE scheme* $\Pi = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ *is* correct *for some class of circuits $\mathscr{L}$ if for all $m_1, \ldots, m_k \in \mathscr{M}$, for all $C \in \mathscr{L}$, for all* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^c)$*, we have that*

$$\mathsf{Dec}_{\mathsf{sk}}(\mathsf{Eval}_{\mathsf{pk}}(C, \mathsf{Enc}_{\mathsf{pk}}(m_1), \ldots, \mathsf{Enc}_{\mathsf{pk}}(m_k))) = C(m_1, \ldots, m_k). \tag{4.2}$$

One can relax the notion of correctness to *statistical* correctness, where the above identity only holds with high probability (over the random coins of Enc and Eval). We will not make a distinction between these two notions.

The work [20] introduced an "approximate" FHE scheme (CKKS), for which Equation (4.2) does not hold. The security implications of this relaxation are investigated in [52], as discussed below. However, neither [20] nor [52] provide a formal definition of an "approximate" FHE scheme, and instead simply drop the correctness requirement (4.2) without any further restriction. This is despite the CKKS scheme satisfying an approximate version of the correctness property of Equation (4.2).

The definition of *approximately correct* FHE scheme plays a fundamental role in our work.

Informally, an approximately correct FHE scheme allows for meaningful, but inexact, computation on encrypted messages. To formalize the relaxed correctness requirements of an approximately correct FHE scheme, we first define the *plaintext error*[2], which specifies the extent to which a homomorphic computation fails to be exact.

Definition 36 (Plaintext Error)**.** *Let* $\Pi = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ *be an FHE scheme with message space* $\mathscr{M} \subseteq \widetilde{\mathscr{M}}$*, which is a normed space with norm* $\|\cdot\| : \widetilde{\mathscr{M}} \to \mathbb{R}_{\geq 0}$*. For any ciphertext* ct*, secret key* sk*, and message m, the* plaintext error *of* $(\mathsf{ct}, m, \mathsf{sk})$ *is defined to be*

$$\mathsf{Error}(\mathsf{ct}, m, \mathsf{sk}) = \|\mathsf{Dec}_{\mathsf{sk}}(\mathsf{ct}) - m\|. \tag{4.3}$$

Typically, for some circuit $C \in \mathscr{L}$, key pair $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^c)$, and input values $m_1, \ldots, m_k \in \mathscr{M}$, one is interested in the quantity $\mathsf{Error}(\mathsf{ct}, m, \mathsf{sk})$ for

$$m = C(m_1, \ldots, m_k), \quad \text{and,} \quad \mathsf{ct} = \mathsf{Eval}_{\mathsf{pk}}(C, \mathsf{Enc}_{\mathsf{pk}}(m_1), \ldots, \mathsf{Enc}_{\mathsf{pk}}(m_k)),$$

i.e. where m and ct correspond to the same computation done on plaintexts and ciphertexts.

In this work we investigate two distinct correctness properties for approximate homomorphic encryption. The first is implicit in the literature on CKKS. We call this notion "static" to contrast with a notion we investigate later in Section 4.5.

Definition 37 (Static Approximate Correctness)**.** *Let* Π *be an FHE scheme with message space* $\mathscr{M} \subseteq \widetilde{\mathscr{M}}$*, which is a normed space with norm* $\|\cdot\| : \widetilde{\mathscr{M}} \to \mathbb{R}_{\geq 0}$*. Let* $\mathscr{L}$ *be a space of circuits,* $\mathscr{L}_k \subseteq \mathscr{L}$ *the subset of parity k circuits, and let* $\mathsf{Estimate} : \bigsqcup_{k \in \mathbb{N}} \mathscr{L}_k \times \mathbb{R}_{\geq 0}^k \to \mathbb{R}_{\geq 0}$ *be an efficiently computable function. We call the tuple* $\tilde{\Pi} = (\Pi, \mathsf{Estimate})$ *a* statically approximate *FHE scheme if for all* $k \in \mathbb{N}$*, for all* $C \in \mathscr{L}_k$*, for all* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^c)$*, if* $\mathsf{ct}_1, \ldots, \mathsf{ct}_k$ *and* $m_1, \ldots, m_k$ *are such that* $\mathsf{Error}(\mathsf{ct}_i, m_i, \mathsf{sk}) \leq t_i$*, then*

$$\mathsf{Error}(\mathsf{Eval}_{\mathsf{pk}}(C, \mathsf{ct}_1, \ldots, \mathsf{ct}_k), C(m_1, \ldots, m_k), \mathsf{sk}) \leq \mathsf{Estimate}(C, t_1, \ldots, t_k).$$

[2]Note that this is different than the *ciphertext* error of Chapter 3.

Note that the type signature $\bigsqcup_{k\in\mathbb{N}} \mathscr{L}_k \times \mathbb{R}_{\geq 0}^k \to \mathbb{R}_{\geq 0}$ encodes that $\mathsf{Estimate}$ takes as input a circuit C, and an error bound t_i for each of the k input wires to the circuit $C \in \mathscr{L}_k$. This correctness notion is "static" in the sense of static typing. In particular, $\mathsf{Estimate}$ only depends on

- the computation C to be done, and
- error bounds t_i for the inputs to the homomorphic computation.

All of these quantities are publicly computable given an abstract description of a computation, and (for non-adaptive computations) can even be precomputed (say by an FHE "compiler").

Generally $\mathsf{Estimate}(\cdot)$ either computes a (provable) worst-case bound on the error, or a (heuristic) average-case bound. Our work assumes worst-case bounds (although we discuss average-case bounds some in Section 4.6). Approximate FHE schemes often require that all $m_1, \ldots, m_k$ are of bounded norm — this can be captured in the above definition by choosing $\mathscr{M}$ to be a set of bounded norm.

Security

We use the following security definition, proposed in [52], which properly captures security of approximate FHE schemes against passive attacks.

Algorithm 1. Oracles for the IND-CPA$^{\mathsf{D}}$ game.

initialization

$(\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^c)$

global state

$S \leftarrow \emptyset$

$i \leftarrow 0$

$\mathsf{E}^b_{\mathsf{pk}}(m_0,m_1) :=$

$\mathsf{ct} \leftarrow \mathsf{Enc}_{\mathsf{pk}}(m_b)$

$S[i] \leftarrow (m_0,m_1,\mathsf{ct})$

$i \leftarrow i+1$

return ct

$\mathsf{H}^b_{\mathsf{pk}}(g,\mathbf{J}=(j_1,\ldots,j_k)) :=$

$\mathsf{ct} \leftarrow \mathsf{Eval}_{\mathsf{pk}}(g,S[j_1].\mathsf{ct},\ldots,S[j_k].\mathsf{ct})$

$gm_0 \leftarrow g(S[j_1].m_0,\ldots,S[j_k].m_0)$

$gm_1 \leftarrow g(S[j_1].m_1,\ldots,S[j_k].m_1)$

$S[i] \leftarrow (gm_0,gm_1,\mathsf{ct})$

$i \leftarrow i+1$

return ct

$\mathsf{D}^b_{\mathsf{sk}}(i) :=$

if $S[i].m_0 = S[i].m_1$

return $\mathsf{Dec}_{\mathsf{sk}}(S[i].\mathsf{ct})$

else

return $\perp$

Definition 38 (IND-CPA$^{\mathsf{D}}$ Security, [52])**.** *Let* $\Pi = (\mathsf{KeyGen},\mathsf{Enc},\mathsf{Dec},\mathsf{Eval})$ *be a FHE scheme. We define the IND-CPA^D game to be an indistinguishability game parameterized by distribution ensembles* $\{(\mathsf{E}^b_\theta,\mathsf{H}^b_\theta,\mathsf{D}^b_\theta)\}_\theta$ *for* $b \in \{0,1\}$*, where these oracles are the (stateful*[3]*) oracles given in Algorithm 1.*

We will use the formalism of (c,s)-bit security introduced in Chapter 2. We remind the reader that we will say that a scheme Π has (c,s)-bits of IND-CPA$^{\mathsf{D}}$ security if for any adversary A either

$$c \leq \log_2 \frac{T(A)}{\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A)}, \qquad \text{or} \qquad s \leq \log_2 \frac{1}{\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A)},$$

where $\mathscr{X}$ is the distinguishing game, where the adversary gets access to the oracles defined in Algorithm 1. We will additionally refer to c-bit security throughout this chapter, by which we mean

[3]As a standard convention (for this and other games defined in the paper), if at any point in a game the adversary makes an invalid query (e.g., a circuit g not supported by the scheme, or indices out of range), the oracle simply returns an error symbol $\perp$.

(c,∞)-bit security, or that any adversary A satisfies $c \leq \log_2 \frac{T(A)}{\mathsf{adv}^{\mathsf{MW}}_{\mathscr{Z}}(A)}$.

In [52] it is also shown that for FHE schemes satisfying the standard correctness requirement (Eq. 4.2), IND-CPA$^{\mathsf{D}}$ security is equivalent to the traditional formulation of indistinguishability under chosen plaintext attack (IND-CPA), defined as follows.

Definition 39 (IND-CPA Security)**.** *Let* $\Pi = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ *be a FHE scheme. We define the* IND-CPA *game to be an indistinguishability game parameterized by distribution ensembles* $\{\mathsf{E}^b_\theta\}_\theta$ *for* $b \in \{0,1\}$ *of Algorithm 1.*

We will additionally use weaker and stronger variants of IND-CPA$^{\mathsf{D}}$, informally defined as follows:

- q-IND-CPA$^{\mathsf{D}}$ security. This is the same as IND-CPA$^{\mathsf{D}}$ security, but restricted to adversaries that make at most $q(c)$ queries to oracle D.
- KR$^{\mathsf{D}}$ security, or security against key recovery attacks. Here we modify the IND-CPA$^{\mathsf{D}}$ game by restricting[4] the E oracle to queries of the form $\mathsf{E}(m,m)$, and requiring the adversary to output (at the end of the attack) a secret key sk', rather than the bit b'. The attack is successful if $\mathsf{sk} = \mathsf{sk}'$, and the advantage of an adversary is measured by the success probability of the adversary.

KR$^{\mathsf{D}}$ security is implied by IND-CPA$^{\mathsf{D}}$ security, but it is much weaker, and it is not generally considered a satisfactory notion of security. Here (as in [52]), KR$^{\mathsf{D}}$ security is used exclusively to show that certain schemes are not secure, making the insecurity results stronger.

4.3 A Differentially Private Approach to IND-CPA$^{\mathsf{D}}$ Security

In this section we investigate achieving q-IND-CPA$^{\mathsf{D}}$ security for statically approximate, IND-CPA-secure FHE schemes $\tilde{\Pi}$. Our approach is to post-process decryptions of $\tilde{\Pi}$ with an appropriate notion of differential privacy. The noise added by this differentially private mechanism

[4]This is without loss of generality, as the only point of general queries $\mathsf{E}(m,m')$ is to get information correlated with the secret bit b, which is not present in this game.

will suffice to information-theoretically hide the plaintext error, allowing us to reduce our analysis to the case of exact FHE, where IND-CPA and q-IND-CPA$^{\mathsf{D}}$ security are equivalent.

4.3.1 Our Notion of Differential Privacy

Our notion of differential privacy is a generalization of the notion of Rényi differential privacy [63] to different norms[5]. As the tightest bounds in our setting occur in the simplest[6] case when $\alpha = 1$, we present things solely in terms of this Rényi divergence, i.e. the KL divergence.

Definition 40 (Norm KL Differential Privacy)**.** *For $t \in \mathbb{R}_{\geq 0}$, let $M_t : B \to C$ be a family of randomized algorithms, where B is a normed space with norm $\|\cdot\| : B \to \mathbb{R}_{\geq 0}$. Let $\rho \in \mathbb{R}$ be a privacy bound. We say that the family M_t is ρ-KL differentially private (ρ-KLDP) if, for all $x, x' \in B$ with $\|x - x'\| \leq t$,*

$$D(M_t(x)||M_t(x')) \leq \rho. \tag{4.4}$$

Note that our mechanism M depends on a bound on the distance $\|x - x'\| \leq t$, which it uses (internally) to set parameters to meet the desired privacy bound. In the most common case of Gaussian noise, it will use noise of standard deviation $\sigma = \Omega(2^{s/2}t)$ to achieve (c,s)-bit security (Theorem 17), e.g. $\sigma = \Omega(2^{c/s}t)$ for c-bits of computational security.

As $\|x - x'\| = \|x' - x\|$ is itself symmetric, our definition is invariant under replacing $D(\mathscr{D}_0||\mathscr{D}_1)$ with $\max(D(\mathscr{D}_0||\mathscr{D}_1), D(\mathscr{D}_1||\mathscr{D}_0))$, and is therefore implicitly dependent on this larger (symmetric) measure, although we do not make this explicit in our work.

[5] In Differential Privacy, "adjacent" values are typically measured in the Hamming norm, while for our purposes the ℓ_2 and ℓ_∞ norms are of primary interest.

[6] There is an alternative simplification of the Rényi divergence when $\alpha = \infty$ known as the *max-log distance* [60] with desirable properties, for example it is a metric, similarly to the statistical distance. As our bounds degrade linearly in α as $\alpha \to \infty$, this notion is unsuitable for our situation.

Algorithm 2. The FHE Scheme $M[\tilde{\Pi}]$

$\mathsf{Enc}'_{\mathsf{pk}}(m) :=$

  $c \leftarrow \mathsf{Enc}_{\mathsf{pk}}(m)$

  return $\mathsf{ct} = (c, t_e)$

$\mathsf{Eval}'_{\mathsf{pk}}(C, \mathsf{ct}'_1, \ldots, \mathsf{ct}'_k) :=$

  $c \leftarrow \mathsf{Eval}_{\mathsf{pk}}(C, \mathsf{ct}_1.c, \ldots, \mathsf{ct}_k.c)$

  $t \leftarrow \mathsf{Estimate}(C, \mathsf{ct}_1.t, \ldots, \mathsf{ct}_k.t)$

  return $\mathsf{ct} = (c, t)$

$\mathsf{Dec}'_{\mathsf{sk}}(\mathsf{ct}) :=$

  return $M_{\mathsf{ct}.t}(\mathsf{Dec}_{\mathsf{sk}}(\mathsf{ct}.c))$

Definition 41. *Let* $\Pi = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ *be an FHE scheme with plaintext space* $\mathscr{M} \subseteq \widetilde{\mathscr{M}}$*, where* $\widetilde{\mathscr{M}}$ *is a normed space with norm* $\|\cdot\|$*. Let* $\mathsf{Estimate}$ *be such that* $\tilde{\Pi} = (\Pi, \mathsf{Estimate})$ *is statically approximate, and let* t_e *be an upper bound on plaintext errors of fresh encryptions* $\mathsf{Enc}_{\mathsf{pk}}(m)$ *for all* $m \in \mathscr{M}$*. Let* M_t *be a* ρ*-KLDP mechanism on* $\widetilde{\mathscr{M}}$*. Define the FHE scheme* $M[\tilde{\Pi}]$ *that has an identical* KeyGen *algorithm to* Π*, with the modified algorithms* $\mathsf{Enc}'_{\mathsf{pk}}, \mathsf{Eval}'_{\mathsf{pk}}$*, and* $\mathsf{Dec}'_{\mathsf{sk}}$ *of Algorithm 2.*

In the above definition of the scheme $M[\tilde{\Pi}]$, we use the "tagged ciphertext" notation $\mathsf{ct} = (c, t)$, where c is an ordinary ciphertext and t is an estimated plaintext error upper bound. An initial estimation t_e is provided by the encryption algorithm, and the evaluation algorithm updates the error upper bound using $\mathsf{Estimate}(\cdot)$ such that the resulting scheme is a statically approximate FHE scheme.

Algorithm 3. The decryption oracle for the game $\mathscr{G}_1$ of Theorem 16.

$\mathsf{D}(i) :=$

  if $S[i].m_0 = S[i].m_1$

    $t_i \leftarrow S[i].\mathsf{ct}.t$

    return $M_{t_i}(S[i].m_0)$

  else

    return $\perp$

Theorem 16. *Let* $\Pi = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ *be an FHE scheme with plaintext space* $\mathscr{M} \subseteq \widetilde{\mathscr{M}}$*, where* $\widetilde{\mathscr{M}}$ *is a normed space with norm* $\|\cdot\|$*. Let* $\mathsf{Estimate}$ *be such that* $\tilde{\Pi} = (\Pi, \mathsf{Estimate})$ *is statically approximate. Let* $c > 0$*, let* M_t *be a* ρ*-KLDP mechanism on* $\widetilde{\mathscr{M}}$ *where* $\rho \leq 2^{-s}/(2q)$*, and let* $q \in \mathbb{N}$*. If* Π *is* (c,s)*-bit secure in the* $\mathsf{IND\text{-}CPA}$ *game, then* $M[\tilde{\Pi}]$ *is* $(c - \log_2 6 - \log_2 T_{\mathscr{G}_2}, s - \log_2 6)$*-bit secure in the q-*$\mathsf{IND\text{-}CPA}^{\mathsf{D}}$ *game, where* $T_{\mathscr{G}_2}$ *is the relative cost of simulation of an explicit game.*

We take slight care in our proof, and show that, when analyzing a particular adversary A, one may replace $T_{\mathscr{G}_2} = \sup_A \frac{T^A_{\mathscr{G}_2}}{T_A}$ with this fraction (without the sup). Note that generally the class of A that one is concerned an adversary may employ to attack computational bit security satisfy $T_A \ll T^A_{\mathscr{G}_2}$, so $\log_2 \frac{T^A_{\mathscr{G}_2}}{T_A} \approx 0$. We will make this assumption in our applications of this result later.

We also repeat (parts of) the proofs of Theorems 8 and 9, as it allows us to mildly optimize constants[7].

Proof. We define a sequence of distinguishing games.

- $\mathscr{G}_0$, the scheme $M[\tilde{\Pi}]$ in the q-$\mathsf{IND\text{-}CPA}^{\mathsf{D}}$ game,
- $\mathscr{G}_1$, the scheme $M[\tilde{\Pi}]$ in the q-$\mathsf{IND\text{-}CPA}^{\mathsf{D}}$ game, with the modified decryption oracle of Algorithm 3, and
- $\mathscr{G}_2$, the scheme $M[\tilde{\Pi}]$ in the $\mathsf{IND\text{-}CPA}$ game, and
- $\mathscr{G}_3$, the scheme Π in the $\mathsf{IND\text{-}CPA}$ game.

By assumption, Π is (c,s)-bit secure in game $\mathscr{G}_3$. Note that $M[\tilde{\Pi}]$ and Π only differ in their decryption oracle, which is not an oracle present in the $\mathsf{IND\text{-}CPA}$ game, so $M[\tilde{\Pi}]$ is easily (c,s)-bit secure in game $\mathscr{G}_2$. Next, note that games $\mathscr{G}_2$ and $\mathscr{G}_1$ only differ in the output of the oracle D (which again, is not present in game $\mathscr{G}_2$, but may be perfectly simulated by an adversary in this game at

[7]These theorems were phrased entirely in terms of $\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A)$, to give clean, self-contained statements. This requires applying bounds $\Delta_{\mathsf{LC}}(A(\mathscr{X}_0), A(\mathscr{X}_1)) \leq 3\mathsf{adv}^{\mathsf{MW}}_{\mathscr{X}}(A^z)$ that incur a mild overhead (multiplicative factor of 3), which we avoid in the current section.

mild[8] running-time cost $T^A_{\mathscr{G}_2}$). It follows that if A' is any adversary against $\mathscr{G}_2$, that

$$\mathsf{adv}^{\mathsf{MW}}_{\mathscr{G}_1}(A') \leq \max(T_{A'} 2^{-c}(1 + \frac{T^{A'}_{\mathscr{G}_2}}{T_{A'}}), 2^{-s}).$$

In what follows, let G^b_i denote the output distribution of A (another arbitrary adversary) in game $\mathscr{G}_i$'s left ($b = 0$) or right ($b = 1$) world. We can compute

$$\begin{aligned}
\sqrt{\mathsf{adv}^{\mathsf{MW}}_{\mathscr{G}_0}(A)} &\leq \Delta_{\mathsf{LC}}(G^0_0, G^1_0) \\
&\leq \Delta_{\mathsf{LC}}(G^0_0, G^0_1) + \Delta_{\mathsf{LC}}(G^0_1, G^1_1) + \Delta_{\mathsf{LC}}(G^1_1, G^1_0) \\
&\leq \Delta_{\mathsf{LC}}(G^0_0, G^0_1) + \Delta_{\mathsf{LC}}(G^1_0, G^1_1) + \sqrt{3\mathsf{adv}^{\mathsf{MW}}_{\mathscr{G}_1}(A')},
\end{aligned}$$

where $A' = A^z$ is the adversary of Lemma 10, which has $T_{A'} \approx T_A$. Recall that $\Delta_{\mathsf{LC}}(G^b_0, G^b_1)$ measures the divergence between

- the output distribution of A during execution in $\mathscr{G}_0$ (G^b_0), and
- the output distribution of A during execution in $\mathscr{G}_1$ (G^b_1).

One may therefore appeal to Lemma 23 (and the fact that M is a ρ-KLDP mechanism) to get the bound

$$\Delta_{\mathsf{LC}}(G^0_0, G^1_1) + \Delta_{\mathsf{LC}}(G^1_0, G^1_1) \leq 2\sqrt{\frac{q\rho}{2}} = \sqrt{2q\rho}.$$

We finally have that

$$\sqrt{\mathsf{adv}^{\mathsf{MW}}_{\mathscr{G}_0}(A)} \leq \sqrt{2q\rho} + \sqrt{\max(T_A 2^{-c}(1 + \frac{T^{A'}_{\mathscr{G}_2}}{T_{A'}}), 2^{-s})} \leq \sqrt{3\max(T_A 2^{-c+2}(1 + \frac{T^{A'}_{\mathscr{G}_2}}{T_{A'}}), 2^{-s+2})},$$

e.g. $\mathscr{G}_0$ is $(c - \log_2 6 - \log_2(1 + T_{\mathscr{G}_2}), s - \log_2 6)$-bit q-$\mathsf{IND\text{-}CPA}^{\mathsf{D}}$ secure. □

[8] In general, simulation requires performing (in plaintext) the computations $\{C_i\}_i$ that were queried to H, and computing M q times, e.g. has additive overhead $T^A_{\mathscr{G}_2} = qT_M + \sum_i T_{C_i}$.

4.3.2 Gaussian Mechanism

In this section, we present and analyze a differentially private mechanism M_t which simply adds Gaussian noise to its input.

Definition 42. *Let $\mu \in \mathbb{Z}$, and $\sigma > 0$. The discrete Gaussian of parameters μ, σ (written $\mathscr{N}_{\mathbb{Z}}(\mu, \sigma^2)$) is the probability distribution supported on $\mathbb{Z}$ with p.m.f. $p(x) \propto \exp(-(x-\mu)^2/2\sigma^2)$.*

It is known how to (with high probability) exactly sample from this distribution in constant time [14]. We explicitly bound the impact of this on the security of our constructions in the full version of our paper.

Proposition 8 (Prop. 5 of [14])**.** *Let $\sigma \in \mathbb{R}_{\geq 0}$, and let $\mu, \nu \in \mathbb{Z}$. Then:*

$$D(\mathscr{N}_{\mathbb{Z}}(\mu, \sigma^2) || \mathscr{N}_{\mathbb{Z}}(\nu, \sigma^2)) = \frac{(\nu - \mu)^2}{2\sigma^2}. \tag{4.5}$$

Definition 43. *Let $\rho > 0$, and $n \in \mathbb{N}$. Define the* (discrete) Gaussian Mechanism $M_t : \mathbb{Z}^n \to \mathbb{Z}^n$ *be the mechanism that, on input $x \in \mathbb{Z}^n$, outputs a sample from $\mathscr{N}_{\mathbb{Z}^n}(x, \frac{t^2}{2\rho} I_n)$.*

Lemma 31. *For any $\rho > 0, n \in \mathbb{N}$, the Gaussian mechanism is ρ-KLDP.*

Proof. Let $\mathscr{X} = \mathscr{N}_{\mathbb{Z}^n}(x, \frac{t^2}{2\rho} I_n)$ and $\mathscr{Y} = \mathscr{N}_{\mathbb{Z}^n}(y, \frac{t^2}{2\rho} I_n)$. By sub-additivity of the KL divergence and Proposition 8, we have that $D(\mathscr{X} || \mathscr{Y}) \leq \|\widehat{D}(\mathscr{X} || \mathscr{Y})\|_1 = \frac{\rho}{t^2} \|x - y\|_2^2 \leq \rho$. □

Corollary 12. *Let $\Pi = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ be an FHE scheme with plaintext space $\mathscr{M} \subseteq \widetilde{\mathscr{M}}$, where $\widetilde{\mathscr{M}} \subseteq \mathbb{Z}^n$ is a normed space with norm $\|\cdot\|$. Let $\mathsf{Estimate}$ be such that $\tilde{\Pi} = (\Pi, \mathsf{Estimate})$ is a statically approximate FHE scheme. Let M_t be the Gaussian mechanism (with $\rho := 2^{-s}/(12q)$). If Π is $(c + \log_2 6, s + \log_2 6)$-bit secure in the $\mathsf{IND\text{-}CPA}$ game, then $M[\tilde{\Pi}]$ is (c, s)-bit secure in the q-IND-CPA^D game.*

Proof. Immediate application of Theorem 16, under the heuristic assumption that $\log_2 T_{\mathscr{G}_2} \approx 0$. □

As the Gaussian mechanism adds noise of standard deviation $\mathsf{ct}.t/\sqrt{2\rho}$ to each coordinate, to attain (c, s)-bit security one loses $s/2 + \log_2 6 + \log_2 \sqrt{q} + \log_2 \mathsf{ct}.t$ bits of precision. As the

ciphertext already contains $\log_2 \mathsf{ct}.t$ bits of noise, the *additional* precision lost by $M[\tilde{\Pi}]$ is $s/2 + \log_2 \sqrt{q} + \log_2 6$ bits.

Proof. This immediately follows from Lemma 31 with Theorem 16. □

This transformation does not explicitly depend on the underlying parameters of the particular implementation of approximate encryption (for example, the size of the LWE moduli one is working over, the dimension of the message space, etc.), and instead only implicitly depends on these quantities via the computation of the static plaintext error bound. We caution that to apply this result to CKKS one needs to be slightly careful about the underlying norm one is working with, which we do later in Theorem 17.

4.4 Application to CKKS

Prior work of [52] shows that the approximate FHE scheme of [20] does not satisfy $\mathsf{IND\text{-}CPA^D}$-security, even though it satisfies $\mathsf{IND\text{-}CPA}$-security. We refer the reader to [52] for additional details, but at a high level they show that publishing the results of an approximate FHE computation under CKKS leaks information about the secret key, enabling a full key recovery attack in the case of trivial computation, and an attack against $\mathsf{IND\text{-}CPA^D}$-security for more general homomorphic computation. In this section, we apply Theorem 16 and Lemma 31 to give a modification of the CKKS decryption function that allows us to prove $\mathsf{IND\text{-}CPA^D}$-security of the modified scheme.

We use the results of Section 4.3 to show that post-processing the results of the CKKS decryption function with the Gaussian mechanism is sufficient to achieve $\mathsf{IND\text{-}CPA^D}$-security for the CKKS scheme, for large enough Gaussian noise (Section 4.4.2). We also prove a nearly matching lower bound on the Gaussian noise necessary to achieve $\mathsf{IND\text{-}CPA^D}$-security for the CKKS scheme (Section 4.4.3). We then briefly examine the countermeasures adopted by some open-source implementations of CKKS, and we suggest concrete parameters (Section 4.4.4).

4.4.1 The CKKS Approximate FHE Scheme

We present the relevant subroutines of the CKKS FHE scheme. We omit many details of the CKKS scheme, and refer the reader to [20] for a more complete description. The CKKS scheme is parameterized by a plaintext dimension $n/2$ (typically a power-of-two), a ciphertext modulus Q, and a discrete Gaussian error distribution χ_σ with standard deviation σ. Complex vectors in $\mathbb{C}^{n/2}$ are considered as messages in CKKS, and they are encoded to plaintext polynomials in R by composing π^{-1} and τ^{-1} together with a scaling factor; conversely, plaintexts are decoded using $\varphi := \tau \circ \pi$, again with a scaling factor. We define the *canonical embedding norm* $\|\cdot\|_\infty^{\mathsf{can}}$ of an element $a \in \mathbb{R}[X]/(\Phi_N(X))$ to be $\|a\|_\infty^{\mathsf{can}} = \|\tau(a)\|_\infty$. We will use this norm to track the plaintext error of CKKS ciphertexts.

- $\mathsf{CKKS.KeyGen}(1^c)$: Take $w = w(c)$ and $p = p(c,Q)$. To generate the secret key sk, sample $s \leftarrow \{s \in \{-1,0,1\}^n : |s|_0 = w\}$ and take $\mathsf{sk} = (1,s)$. To generate the public key pk, sample $a \leftarrow R_Q$, $e \leftarrow \chi$, and take $\mathsf{pk} = (b = -as + e, a)$. To generate the evaluation key ek, sample $a' \leftarrow R_{pQ}$, $e' \leftarrow \chi$, and take $\mathsf{ek} = (b', a')$ for $b' = -a's + e' + ps^2 \mod pQ$. Return $(\mathsf{sk}, \mathsf{pk}, \mathsf{ek})$.

- $\mathsf{CKKS.encode}(\mathbf{x} \in \mathbb{C}^{n/2}; \Delta)$: Return $\lfloor \Delta \cdot \varphi^{-1}(\mathbf{x}) \rceil \in R$.

- $\mathsf{CKKS.Enc}_{\mathsf{pk}}(m)$: Let T denote the distribution over $\{0, \pm 1\}^n$ induced by sampling each coordinate independently, drawing -1 with probability 1/4, 1 with probability 1/4, and 0 with probability 1/2. Sample $r \leftarrow T$, $e_0, e_1 \leftarrow \chi$, and return $r \cdot \mathsf{pk} + (m + e_0, e_1) \mod Q$.

- $\mathsf{CKKS.Add}(\mathbf{c}_0, \mathbf{c}_1 \in R_Q)$: Return $\mathbf{c}_0 + \mathbf{c}_1 \mod Q$.

- $\mathsf{CKKS.Mult}_{\mathsf{ek}}(\mathbf{c}_0, \mathbf{c}_1 \in R_Q)$: For $\mathbf{c}_0 = (b_0, a_0)$ and $\mathbf{c}_1 = (b_1, a_1)$, let $(b_2, a_2) = (b_0 b_1, a_0 b_1 + a_1 b_0) + \lfloor p^{-1} \cdot a_0 a_1 \cdot \mathsf{ek} \rceil \mod Q$. Return (b_2, a_2).

- $\mathsf{CKKS.decode}(a \in R; \Delta)$: Return $\varphi(\Delta^{-1} \cdot a) \in \mathbb{C}^{n/2}$.

- $\mathsf{CKKS.Dec}_{\mathsf{sk}}(\mathbf{c})$: For $\mathbf{c} = (b, a) \in R_Q^2$, return $b + as \mod Q$.

Note that CKKS supports encryption and decryption of floating-point inputs by pre-processing encryption with CKKS.encode, and post-processing decryption with CKKS.decode. All intermediate operations are then done with integer arithmetic. To simplify exposition, we focus on these intermediate operations, and therefore restrict to the case of integer arithmetic.

We will need the following (standard) expressions for how the error[9] transforms during addition and multiplication.

Lemma 32 (Error Growth [20])**.** *Let $b \in \{0,1\}$, and let $\mathbf{c}_b := (\widehat{c}_{0,0}, \widehat{c}_{0,1}) = \mathsf{CKKS.Enc}_{\mathsf{pk}}(\widehat{m}_b)$ be* CKKS *ciphertexts with errors e_b. Then the ciphertext $\mathbf{c}_{\mathsf{Mult}} = \mathsf{CKKS.Mult}(\mathbf{c}_0, \mathbf{c}_1)$ has error $m_0 e_1 + m_1 e_0 + e_0 e_1 + e_{\mathsf{Mult}}$ for a term e_{Mult} that depends on the parameters of the* CKKS *instance (and the ciphertexts $\mathbf{c}_0, \mathbf{c}_1$). The ciphertext $\mathbf{c}_{\mathsf{Add}} = \mathsf{CKKS.Add}(\mathbf{c}_0, \mathbf{c}_1)$ has error $e_0 + e_1$.*

Certain authors have suggested various heuristics for analyzing e_{Mult}. We will find the following one useful for the analysis of the attack of Section 4.4.3.

Heuristic 2 (Appendix A.5 of [34])**.** *Let w be the hamming weight of* sk*. Then e_{Mult} may be modeled as a random variable with mean zero and variance $O(wn)$.*

The rest of our work will benefit from the following notation.

Definition 44. *For $\sigma > 0$, let $\mathsf{S\text{-}CKKS}_\sigma$ be the* CKKS *encryption scheme, where one modifies decryption to compute $\mathsf{S\text{-}CKKS}_\sigma.\mathsf{Dec}_{\mathsf{sk}}(\mathsf{ct}) = \mathsf{CKKS.Dec}_{\mathsf{sk}}(\mathsf{ct}.c) + \mathscr{N}_{\mathbb{Z}^n}(0, \sigma^2 \mathsf{ct}.t^2 I_n)$.*

4.4.2 IND-CPA$^{\mathsf{D}}$-Secure CKKS

It is straightforward to apply Corollary 12 to CKKS to obtain q-IND-CPA$^{\mathsf{D}}$ security.

Theorem 17. *For any $q \in \mathbb{N}$, if* CKKS *is $(c + \log_2 6, s + \log_2 6)$-bit* IND-CPA*-secure, and $\sigma = 8\sqrt{qn}2^{s/2}$, then $\mathsf{S\text{-}CKKS}_\sigma$ is (c,s)-bit q-IND-CPA$^{\mathsf{D}}$-secure, i.e. $s/2 + \tilde{O}(1)$ additional bits of Gaussian noise suffice to achieve q-IND-CPA$^{\mathsf{D}}$ security.*

[9] Note here, we are being somewhat ambiguous about what CKKS errors are. Precisely, they are *ciphertext errors*, though for CKKS, these are very closely related to the plaintext errors.

Proof. This follows immediately from Corollary 12, (using the aforementioned inequality $\|m\|_\infty^{\mathsf{can}} \leq \sqrt{n}\|m\|_2$, as our analysis of the Gaussian mechanism uses an ℓ_2 norm bound).

□

4.4.3 Lower Bound for Gaussian Mechanism

Together, Lemma 31 and Theorem 16 give an upper bound on the amount of Gaussian noise required to achieve (c,s)-bits of $\mathsf{IND\text{-}CPA}^{\mathsf{D}}$-security for an $\mathsf{IND\text{-}CPA}$-secure approximate encryption scheme. In this subsection, we show that this upper bound is essentially tight (at least for $c = s$, the setting of computational bit security) for CKKS by demonstrating an attack against $\mathsf{IND\text{-}CPA}^{\mathsf{D}}$ security for noticeably smaller Gaussian noise, i.e. analyzing $\mathsf{S\text{-}CKKS}_{\sigma_s}$ for sanitization noise $\sigma_s \ll 8\sqrt{qn}2^{c/2}$. In what follows, recall that $n = \varphi(N)$, and w denotes the Hamming weight of the key sk.

Algorithm 4. Adversary $A(1^c, \mathsf{pk}, \mathsf{ek})$

for $i \in \{0, \ldots, 44\}$ **do**

$\mathsf{ct}_i \leftarrow \mathsf{E}_{\mathsf{pk}}(m_i^{(0)} = 0, m_i^{(1)} = B)$;

end for

for $i \in \{45, \ldots, 59\}$ **do**

$\mathsf{ct}_i \leftarrow \mathsf{E}_{\mathsf{pk}}(m_i^{(0)} = 0, m_i^{(1)} = -B)$;

end for

$\mathsf{ct}_{60} \leftarrow \mathsf{H}_{\mathsf{ek}}(g, \{0, \ldots, 59\})$ for $g(x_0, \ldots, x_{59}) = \sum_{i=0}^{29}(x_i \cdot x_{30+i})$

$m' \leftarrow \mathsf{D}_{\mathsf{sk}}(60)$

$V_0 = 30\sigma^4 + O(wn) + \sigma_s^2$ *Variance of* $\tau(m')_0$ *if* $b = 0$

$V_1 = 30\sigma^4 + 60B^2\sigma^2 + O(wn) + \sigma_s^2$ *Variance of* $\tau(m')_0$ *if* $b = 1$

if $|\tau(m')_0| < \sqrt{\frac{\log(V_1/V_0)V_0V_1}{V_1 - V_0}}$ **then**

return 0

else

return 1

end if

At a high level, the adversary A will exploit the message-dependence of the S-CKKS error growth (Lemma 32) to design an H query such that the expected magnitude of the plaintext error of ct_{60} is larger when $b = 1$ than when $b = 0$. The adversary A will then query D on this ciphertext, and choose its bit based on the size of the message m' it receives.

Our result below will require the following lower bound on the statistical distance between two Gaussians.

Lemma 33 (Theorem 1.3 [27])**.** *Let* $\sigma_0, \sigma_1 > 0$*. Then*

$$\Delta_{\mathsf{SD}}(\mathcal{N}(0, \sigma_0^2), \mathcal{N}(0, \sigma_1^2)) \geq \frac{1}{200} \min\left\{1, \frac{|\sigma_0^2 - \sigma_1^2|}{\sigma_0^2}\right\}. \tag{4.6}$$

We will next show that the aforementioned adversary will have noticeable advantage unless

σ_s is larger than σ (the standard deviation of the underlying RLWE error) by a factor super-polynomial in the security parameter.

Lemma 34. *Let $\sigma_s > 0$. Then there exists an adversary A against $\mathsf{S\text{-}CKKS}_{\sigma_s}$ in the IND-CPA^D such that $\mathsf{adv}^A = \Omega\left(\frac{1}{\sigma_s^4 n^6}\right)$.*

Proof. We first observe that the ciphertext $\mathsf{ct}_{60} = \mathsf{Eval}_{\mathsf{ek}}(g, \{0,\ldots,59\})$ is an approximate encryption of 0 both when $b = 0$ and $b = 1$ in the $\mathsf{IND\text{-}CPA}^{\mathsf{D}}$ experiment. Therefore the decryption query made by A returns a value rather than $\perp$.

If $b = 0$, then because all ciphertexts ct_i encrypt messages $m_i = 0$, the message-dependent terms of the error growth from Lemma 32 are also 0, and so the plaintext error of ct_{60} is $\sum_{i=0}^{29} e_{\mathsf{Mult}} + e_i e_{30+i}$, where e_i denotes the plaintext error of ct_i. Recall that if error vectors e and e' have entries sampled from a discrete Gaussian with parameter σ, then each of the components of $\tau(ee')$ is distributed with mean 0 and variance σ^4. We can then use the Central Limit Theorem to approximate the distribution of the sum $\sum_{i=0}^{29} e_{\mathsf{Mult}} + e_i e_{30+i}$ as a Gaussian distribution with mean 0 and variance $30\sigma^4 + O(wn)$. Note that this approximation can be improved by increasing the number of terms in the sum to a larger constant. For the sake of concreteness we have designed the adversary such that there are 30 terms, as this is the value at which the Central Limit Theorem is empirically justified.

If $b = 1$, then the message-dependent terms of the error growth are significant, and the error of ct_{60} is

$$\sum_{i=0}^{14} \left(e_{\mathsf{Mult}} + e_i e_{30+i} + B e_i + B e_{30+i}\right) + \sum_{i=15}^{29} \left(e_{\mathsf{Mult}} + e_i e_{30+i} - B e_i + B e_{30+i}\right).$$

As in the case where $b = 0$, we will approximate this distribution as a Gaussian with mean 0. Though the error terms $e_i e_{30+i}$ and $B e_i + B e_{30+i}$ are not independent, they do have covariance 0, as do the terms $e_i e_{30+i}$ and $B e_{30+i} - B e_i$, and so we can approximate the sum of errors as being drawn from a discrete Gaussian distribution with mean 0 and variance $30\sigma^4 + 60B^2\sigma^2 + O(wn)$.

The adversary sees the result of post-processing the error term with the Gaussian mechanism, run with parameter σ_s, and then chooses its bit to return based on the absolute value of the first

component $\tau(m')_0$ under the canonical embedding. When $b = 0$, this means the adversary sees a sample drawn from a distribution that is well-approximated by a centered Gaussian with variance $V_0 = 30\sigma^4 + O(wn) + \sigma_s^2 \mathsf{ct}.t^2$. When $b = 1$, however, the adversary sees a sample drawn from a distribution that is well-approximated by a Gaussian with the same mean, but larger variance $V_1 = 30\sigma^4 + 60B^2\sigma^2 + O(wn) + \sigma_s^2 \mathsf{ct}.t^2$. Let

$$x = \sqrt{\frac{\log(V_1/V_0)V_0V_1}{V_1 - V_0}}.$$

A straightforward calculation shows that for $|\tau(m')_0| < x$, m' is a more likely outcome when $b = 0$ than when $b = 1$, and when $|\tau(m')_0| \geq x$, m' is at least as likely when $b = 1$ as it is when $b = 0$. Then we have that the advantage of adversary A is approximately the total variation distance between a Gaussian with variance V_0 and a Gaussian with variance V_1. By Lemma 33, we have that

$$\Delta(\mathscr{N}(0,V_0), \mathscr{N}(0,V_1)) \geq \frac{1}{200}\frac{|V_0 - V_1|}{V_0} \in \Theta\left(\frac{B^2\sigma^2}{\sigma^4 + wn + \sigma_s^2 \mathsf{ct}.t^2}\right).$$

Recall that w is the hamming weight of the secret key sk, and so we have $w < n$. For security, we know that $\sqrt{n} < \sigma$, and so it follows that the advantage of our (non-aborting) adversary A against the $\mathsf{IND\text{-}CPA}^{\mathsf{D}}$ security of CKKS is the *square* of the total variation distance, i.e. $\Theta\left(\frac{B^4\sigma^4}{(\sigma^4 + \sigma_s^2 \mathsf{ct}.t^2)^2}\right)$. Finally, note that for $\|e_i\|_\infty^{\mathsf{can}} < \sigma n$ holds with high probability, so $\mathsf{ct}.t \leq O(B\sigma n^{3/2})$ (where we pick up a $\sqrt{n}$ factor to convert to the ℓ_2 norm), and therefore the advantage of our adversary is $\Theta\left(\frac{B^4\sigma^4}{\sigma^8 + \sigma_s^4\sigma^4B^4n^6}\right) = \Omega\left(\frac{1}{\sigma_s^4 n^6}\right)$.

□

Theorem 18. *If* $\mathsf{S\text{-}CKKS}_{\sigma_s}$ *is* (c,c)*-bit* $\mathsf{IND\text{-}CPA}^{D}$*-secure, then* $\sigma_s = \Omega(2^{c/4}/n^{3/2})$*, i.e. one must add at least* $c/4 - \tilde{\Omega}(1)$ *bits of additional Gaussian noise.*

Proof. We have that $c \leq \log_2 O\left(\frac{T_A}{\mathsf{adv}_{\mathscr{G}}^{\mathsf{MW}}(A)}\right) \leq \log_2 O(\sigma_s^4 n^6) \implies \sigma_s \geq 2^{c/4}/n^{3/2}$, and therefore $c/4 - \log_2 \Omega(n^{3/2}) \leq \log_2 \sigma_s$.

□

Table 4.1. Additional size of Gaussian noise (measured in bits) required by the countermeasure of Theorem 17 to achieve (c,s)-bits (Definition 13) of q-IND-CPA$^{\mathsf{D}}$-security. Here, q is a bound on the number of decryption queries, and $n \leq 2^{15}$ is a bound on the ring dimension, chosen as it is the highest dimension parameter in the Homomorphic Encryption Standard [2].

$s \backslash q$	1	2^5	2^{10}	2^{15}
128	71.79	74.29	76.79	79.29
112	63.79	66.29	68.79	71.29
96	55.79	58.29	60.79	63.29
80	47.79	50.29	52.79	55.29
64	39.79	42.29	44.79	47.29
48	31.79	34.29	36.79	39.29
32	23.79	26.29	28.79	31.29

We therefore see that while one can potentially improve on the concrete countermeasure of Section 4.4.4, the main (exponential) term is within a constant factor of correct.

A Sample Instantiation

We briefly describe a concrete instantiation of our countermeasure that achieves $(128,64)$-bits of q-IND-CPA$^{\mathsf{D}}$-security. Throughout, we let the number of supported decryption queries be $q = 2^{10}$. Note that one can always (later) support more decryption queries, by rekeying when one runs out. Parameterize CKKS to achieve 131-bits of IND-CPA-security, where $131 > 128 + \log_2 6$. Let n be the resulting dimension of the chosen CKKS instance. We will assume $n \leq 2^{15}$, as every choice of parameters from the Homomorphic Encryption Standard [2] satisfies this bound.

Then, by Theorem 17, if $\sigma = \sqrt{6qn}2^{s/2}$, then S-CKKS$_\sigma$ is (c,s)-bit q-IND-CPA$^{\mathsf{D}}$-secure. In particular, this loses another $s/2 + \log_2\sqrt{6qn}$ bits of precision compared to decrypting via returning $\mathsf{CKKS.Dec}_{\mathsf{sk}}(\mathsf{ct}.c)$. The particular value of $s/2 + \log_2\sqrt{6qn}$ can be found in Table 4.1 as the entry labeled $(s,q) = (64, 2^{10})$, which is 44.79. Therefore, adding an additional 44.79 bits of i.i.d. Gaussian noise suffices to achieve (128, 64)-bits of q-IND-CPA$^{\mathsf{D}}$-security.

4.4.4 Parameters for Concrete Countermeasures

As the attack in [52] was made publicly available, the major open-source implementations of the CKKS scheme adopted several different countermeasures. We briefly summarize these countermeasures in this subsection, and we propose concrete parameters for them to achieve the desired IND-CPA$^{\mathsf{D}}$ security.

HElib.

The decryption API implementation was modified to add pseudorandom Gaussian noise to the raw decryption result. By default, HElib implements S-CKKS$_1$, e.g. the size of the extra noise is equal to the size of the static error bound of the homomorphic computation. HElib also provides an optional precision parameter in its decryption API such that the extra noise is chosen to be the largest within the precision requirement (for example, if the static error bound is not tight). To achieve (c,s)-bit security against at most $q \geq 1$ decryption queries, this precision parameter should be calibrated such that sufficient (as quantified in Theorem 17 and Table 4.1) noise is added during decryption.

HEAAN, Lattigo.

These libraries require the default decryption API to be used only by the secret key holder, and they added a specialized decryption API to share the decryption results publicly. In HEAAN, the new decryption API takes a noise size parameter, which sets the amount of Gaussian noises to be added to the raw decryption result. In Lattigo, the new decryption API takes a rounding parameter, which is used to round the raw decryption result to certain precision. For both of them, one must estimate the plaintext error $\mathsf{ct}.t$ separately and set the noise parameter as in Theorem 17 and Table 4.1 to achieve (c,s)-bit security against q decryptions.

PALISADE.

The decryption function in PALISADE also adds Gaussian noise to the raw decryption result, but the size of the noise is chosen (dynamically) in a way detailed in Section 4.5.

4.4.5 The Impact of Our Countermeasure

Evaluating the feasibility of our countermeasure for some application depends on both the required (application) precision, as well as the supported (library) precision. Provided the difference between these is larger than the sum of the DP noise (as measured in Table 4.1) with the approximation error, our countermeasure should be able to be instantiated.

32-bit applications.

Concretely, many applications (say in machine learning) require 32 bits of precision. If a FHE library only supports computations with up to 64 bits of precision, this leaves at most 32 bits available for the sum of the CKKS approximation error and the DP error induced by our countermeasure. This means that at best, one will be able to choose $s \approx 32$, which is likely too low for most applications. Note that if the FHE library supports up to 128-bit precision computations[10], this problem disappears, as there are now ≈ 96 bits available for the sum of the errors, allowing the conservative choice of $s \approx 128$.

Low-precision applications.

Some applications may solely require 8 or 16 bits of precision (see for example [44] or [83] for work on training ML models with low-precision computations). This leaves 48-56 bits of precision for the sum of the CKKS approximation error and the DP error. One can then choose $s \approx 64$ (16-bit required precision) or 80 (8-bit), where precise choices of s would depend on the size of the CKKS approximation error. We view either of these choices as much more reasonable than $s \approx 32$, although in all settings the particular choice of s that is appropriate is application-dependent.

4.5 Dynamic Error Estimation

Yuriy Polyakov [71] has recently suggested a technique to get sharper bounds on the plaintext error of the CKKS scheme. Briefly, this is done via leveraging a special message encoding which fixes many of the coordinates of the original CKKS message space to be constantly 0. Provided

[10] At the time of writing, Lattigo and PALISADE already supported computations of this precision for unrelated reasons, e.g. this is a reasonable assumption.

one only evaluates functions which ignore these coordinates, upon decryption these coordinates will only contain the error incurred during the homomorphic computation, and one can attempt to generalize the (exact) error measurements within these coordinates to an estimate of the entirety of the error.

This notion differs from our notion of static approximate correctness in two significant ways, namely

- it depends on the particular ciphertext one is estimating the error of, e.g. can only be computed *dynamically* during the program "run-time", and
- it can only be computed during decryption, e.g. is not *publicly-computable* information about the ciphertext.

We investigate the IND-CPA$^{\mathsf{D}}$ security of applying our transformation of Definition 41 to an approximate encryption scheme that is correct in the "dynamic" sense sketched above. In this slightly modified setting, we get significantly different results. For an IND-CPA-secure, dynamic approximately correct FHE scheme $\tilde{\Pi}$, we find that $M[\tilde{\Pi}]$ is often insecure. Specifically, assuming a "non-triviality" condition on M that we define in Definition 47, we find that

1. for a "natural" class of IND-CPA-secure $\tilde{\Pi}$ (including CKKS), $M[\tilde{\Pi}]$ is not q-IND-CPA$^{\mathsf{D}}$ secure when one uses dynamic error estimation, and
2. there exists an IND-CPA-secure $\tilde{\Pi}$ such that $M[\tilde{\Pi}]$ is not KR$^{\mathsf{D}}$-secure (again, when one uses dynamic error estimation).

4.5.1 A (Heuristic) Dynamic Estimation Procedure for CKKS

We first provide a detailed description of Yuriy Polyakov's dynamic error estimation procedure for CKKS [71], which has been implemented in PALISADE [64]. We define a variant DE-CKKS of CKKS that is modified to use this dynamic error estimation technique. The message space of DE-CKKS is the set of real vectors $\mathbb{R}^{n/2}$, which is a subset of the message space $\mathbb{C}^{n/2}$ of

CKKS. We use $\Re(z)$ and $\Im(z)$ to denote the real and imaginary parts of a complex number $z \in \mathbb{C}$, respectively. We now describe the modified scheme DE-CKKS.

- DE-CKKS.KeyGen: The parameter and key generation algorithms are identical to CKKS, except that the conjugation keys are not generated anymore.
- DE-CKKS.encode: The encoding algorithm is the same as in CKKS, except that it takes only real vectors $\mathbf{x} \in \mathbb{R}^{n/2}$.
- DE-CKKS.Enc: The encryption algorithm is identical to CKKS.
- DE-CKKS.Eval: The homomorphic evaluation algorithm is also identical to CKKS, except that homomorphic conjugation operation is no longer supported.
- DE-CKKS.Dec: The modified decryption algorithm combines the decryption and decoding algorithms of CKKS, and it works as follows given the secret key sk and a ciphertext ct.
  1. Decrypt ct and then decode the vanilla CKKS decryption result: $\mathbf{z} = \mathsf{CKKS.decode}(\mathsf{CKKS.Dec}_{\mathsf{sk}}(\mathsf{ct}))$. Note that $\mathbf{z} \in \mathbb{C}^{n/2}$ is a complex vector.
  2. Let $\mathbf{x} = \Re(\mathbf{z})$, and $\mathbf{e} = \Im(\mathbf{z})$. Estimate the standard deviation $\sigma_e = \mathrm{stdev}(\mathbf{e})$.
  3. Return $\mathbf{x} + \mathbf{r}$, where $\mathbf{r} \leftarrow \mathcal{N}(0, \sqrt{q+1} \cdot \sigma_e I_n)$ is a Gaussian noise vector.

In practice, since the canonical embedding is a scaled isometry with respect to the ℓ_2 norm, we can add the same amount of noise without decoding by first decrypting ct to obtain the ring element $m = \mathsf{CKKS.Dec}_{\mathsf{sk}}(\mathsf{ct})$, computing the ℓ_2 norm of $\frac{1}{2}(m(X) - m(1/X))$ to obtain $\sigma'_e = \sqrt{n} \cdot \sigma_e$, adding $n/2$ i.i.d. Gaussians of parameter $\sqrt{q+1} \cdot \sigma'_e$ to m' and then decoding the resulting noisy ring element.

The PALISADE development team has done some experiments to validate this dynamic error estimation method, and they claimed that it estimates the error well [71]. With optimizations described in [45], they assumed that the rescaling error dominates the plaintext error after each rescaling operation, and that such error can be reduced in size similar to the plaintext error in

fresh encryptions. Furthermore, they assumed the adversary is non-adaptive, meaning that the input messages do not depend on any decryption result. Their experiments encrypted two random real vectors, homomorphically evaluated their component-wise product followed by a rescaling operation, and then decrypted the resulting ciphertext and compared the estimated error size with the actual plaintext error. The results showed that the dynamic error estimation is very close to the actual plaintext error sizes: for example, they differ by at most 2 bits when the lattice dimension is $n = 2^{13}$.

4.5.2 Dynamic Estimation

We next introduce the notion of a dynamically approximately correct FHE scheme $\tilde{\Pi}$. Our notion of dynamic approximate correctness depends on solely the "run-time" values of the FHE scheme, namely the secret key sk, and the ciphertext ct one wishes to bound. These suffice to instantiate the dynamic estimation scheme described in Section 4.5.1. We omit the other values (such as individual plaintext error bounds t_i, and the circuit C itself) for simplicity — there clearly cannot be a security benefit to this omission, as an adversary can easily record or compute these values.

Definition 45 (Dynamic Approximate Correctness)**.** *Let Π be a FHE scheme with message space $\mathscr{M} \subseteq \widetilde{\mathscr{M}}$, which is a normed space with norm $\|\cdot\| : \widetilde{\mathscr{M}} \to \mathbb{R}_{\geq 0}$. Let $\mathscr{L}$ be a space of evaluatable functions, and let $\mathsf{Estimate} : \mathscr{SK} \times \mathscr{C} \to \mathbb{R}_{\geq 0}$ be an efficiently computable function. We call the tuple of algorithms $\tilde{\Pi} = (\Pi, \mathsf{Estimate})$ a* dynamically approximately correct *FHE scheme if for all $m_1, \ldots, m_k \in \mathscr{M}$, for all $C \in \mathscr{L}$, for all $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^c)$, for all $\mathsf{ct} \leftarrow \mathsf{Eval}_{\mathsf{pk}}(C, \mathsf{Enc}_{\mathsf{pk}}(m_1), \ldots, \mathsf{Enc}_{\mathsf{pk}}(m_k))$, we have that*

$$\|\mathsf{Dec}_{\mathsf{sk}}(\mathsf{ct}) - C(m_1, \ldots, m_k)\| \leq \mathsf{Estimate}_{\mathsf{sk}}(\mathsf{ct}). \tag{4.7}$$

The above notion is a "perfect" notion of dynamic approximate correctness — there is an obvious statistical notion as well, where the desired inequality solely has to hold with high probability over all of the various sources of randomness. For simplicity of exposition we will work

with the perfect notion.

We will view the notion of dynamic approximate correctness as a refinement of the notion of static approximate correctness. This can be done without loss of generality, as

- every known approximate FHE scheme is statically correct, and
- the minimum of two (correct) estimation functions is correct.

4.5.3 Attack Against IND-CPA$^{\mathsf{D}}$-Security of $M[\tilde{\Pi}]$ for "Natural" Π

We next attack the IND-CPA$^{\mathsf{D}}$ security of $M[\tilde{\Pi}]$ for "natural" dynamically correct schemes $\tilde{\Pi}$. We briefly summarize the attack, as it is both "obvious", and establishing it theoretically requires a few new definitions (as it fails for "unnatural" schemes). If

- dynamic error estimation is able to tightly estimate the plaintext error,
- the growth of plaintext error during certain operations (such as multiplication) is dependent on the input to the operation, and
- the noise the KLDP mechanism M_t adds is dependent on t in a noticable way, then

an adversary which can distinguish the smaller KLDP noise can immediately break q-IND-CPA$^{\mathsf{D}}$-security. This is simply because one can use the aforementioned operation to construct two ciphertexts $\mathsf{ct}_0, \mathsf{ct}_1$ that encrypt the same value, but have drastically different plaintext errors. Then, as the dynamic error estimation can detect this, the KLDP mechanism will add drastically different noise in the left and right worlds of the q-IND-CPA$^{\mathsf{D}}$ game, immediately breaking security.

The attack is straightforward to implement, which we demonstrate in Section 4.5.4. We next theoretically establish the validity of the attack, by defining the aforementioned notions of "naturality".

Definition 46 (τ-Separated Noise Estimation)**.** *Let $\tilde{\Pi}$ be a dynamically approximately correct FHE scheme with message space $\mathscr{M}$ and space of evaluatable functions $\mathscr{L}$. Let $\tau \geq 1$, and let $C \in \mathscr{L}$ be a circuit. For $m_0, m_1 \in \mathscr{M}$, let $t(m) = \mathsf{Estimate}_{\mathsf{sk}}(\mathsf{Eval}_{\mathsf{pk}}(C, \mathsf{Enc}_{\mathsf{pk}}(m)))$. We say that C has*

τ-separated noise under $\tilde{\Pi}$ if there exists $m_0, m_1 \in \mathscr{M}$ such that $\tau t(m_0) = t(m_1)$ with non-negligible probability.

The seemingly strong condition $t_1 = \tau t_0$ can be replaced by requiring that $|t_0 - \tau t_1|$ is small, and the mechanism M_t produces larger noise as t increases. For example, the Gaussian mechanism adds noise of variance $\sigma^2 = t^2/2\rho$, which increases monotonically with t.

Definition 47 (τ-Sensitivity)**.** *Let M_t be a ρ-KLDP mechanism on a normed space $\mathscr{M}$, and let $\tau : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$. We say that M_t is* τ-sensitive *at $m \in \mathscr{M}$ if for any $t \geq 1$, the distributions $M_t(m) \not\approx_c M_{t\tau}(m)$ are computationally distinguishable.*

The trivial 0-KLDP mechanism (which ignores its input, and returns a fixed constant) is not τ-sensitive for any τ. Note that this condition is desirable in practice — if M_t is not τ-sensitive, there is no real point in getting sharper noise estimates.

Theorem 19. *Let $\tilde{\Pi}$ be an* IND-CPA*-secure, dynamically approximately correct FHE scheme with message space $\mathscr{M}$ and space of evaluatable functions $\mathscr{L}$. Let $\tau : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$, and assume that M is a ρ-KLDP mechanism which is τ-sensitive at 0. Furthermore, assume there exist $m_0, m_1 \in \mathscr{M}$ and $C \in \mathscr{L}$ such that $C(m_0) = C(m_1) = 0$ and C has τ-separated noise estimation under $\tilde{\Pi}$ with respect to inputs m_0, m_1. Then $M[\tilde{\Pi}]$ is not IND-CPA^D-secure.*

The proof is simply a formalization of our sketched attack above, so we omit it.

While it is not clear how to extend this attack to an attack on KR^{D} security (as was present in [52]), the attack still leaks information correlated with $\|m\|$, e.g. breaks semantic security.

4.5.4 Breaking q-IND-CPA$^{\mathsf{D}}$-Security of PALISADE's Dynamic Error Estimation Countermeasure

We implemented the attack in Theorem 19 against the PALISADE's implementation of CKKS, which is currently the only known implementation of dynamic noise estimation. Our attack experiments use the exceedingly simple circuit $f(x_1, x_2) = x_1^2 - x_2$, as well as the circuit $g(x_0, \ldots, x_{4k-1}) = \sum_{i=0}^{2k-1}(x_i \cdot x_{2k+i})$ in Algorithm 4. Notice that both f and g evaluate to 0 on input

Table 4.2. The experimental results of applying the attack in Theorem 19 with various circuits C. Here, $C \in \{f, g\}$, for $f(x_1, x_2) = x_1^2 - x_2$ and $g(x_0, \ldots, x_{4k-1}) = \sum_{i=0}^{2k-1}(x_i \cdot x_{2k+i})$. For both C, denote z_0 the decryption result of $\mathsf{Eval}_{\mathsf{pk}}(C, \mathsf{Enc}_{\mathsf{pk}}(\mathbf{0}))$, and z_m the decryption result of $\mathsf{Eval}_{\mathsf{pk}}(C, \mathsf{Enc}_{\mathsf{pk}}(\mathbf{m}))$ for the input $\mathbf{m}$ as defined above with parameters B and k. We set the lattice parameters (n, Q) to achieve at least 128 bit IND-CPA security, and we choose several different values for the scaling factor Δ and the slots number. For each parameter set, we run the attack 100 times and report the average and standard deviation of $\|z_0\|_\infty$ and $\|z_m\|_\infty$. As shown in the last two columns, there are clear distinctions on the estimated noise sizes between ciphertexts evaluated on $\mathbf{0}$ and $\mathbf{m}$.

C	$(n, \log Q)$	$\log \Delta$	B	k	#slots	$\|z_0\|_\infty$	$\|z_m\|_\infty$
f	$(2^{13}, 100)$	40	100	-	1	$2.19\text{e}-8 \pm 1.83\text{e}-8$	$2.75\text{e}-6 \pm 2.19\text{e}-6$
			100	-	1024	$1.07\text{e}-7 \pm 1.42\text{e}-8$	$1.87\text{e}-5 \pm 2.54\text{e}-6$
			32	-	1	$1.97\text{e}-8 \pm 1.52\text{e}-8$	$1.06\text{e}-6 \pm 1.06\text{e}-6$
			32	-	1024	$1.08\text{e}-7 \pm 1.54\text{e}-8$	$6.08\text{e}-6 \pm 8.85\text{e}-7$
g	$(2^{14}, 150)$	45	32	15	1	$1.08\text{e}-8 \pm 4.37\text{e}-9$	$2.27\text{e}-7 \pm 1.95\text{e}-7$
			32	15	1024	$1.08\text{e}-8 \pm 4.14\text{e}-9$	$1.40\text{e}-6 \pm 2.02\text{e}-7$
			16	50	1	$1.07\text{e}-8 \pm 4.45\text{e}-9$	$2.00\text{e}-7 \pm 1.90\text{e}-7$
			16	50	1024	$1.06\text{e}-8 \pm 4.67\text{e}-9$	$1.27\text{e}-6 \pm 1.70\text{e}-7$

$\mathbf{0}$. On the other hand, we chose several moderate values of $B > 0$ to set the input $\mathbf{m}$ such that $f(\mathbf{m}) = 0$ and $g(\mathbf{m}) = 0$:

- For f, let $m_1 = B$ and $m_2 = B^2$.
- For g, let $m_i = B$ for all $0 \le i \le 3k-1$, and let $m_i = -B$ for all $3k \le i \le 4k-1$.

Our attack homomorphically evaluates f (or g) on encryptions of both $\mathbf{0}$ and $\mathbf{m}$, then it decrypts the final ciphertexts to get z_0 and z_m. As expected, in all our experiments we see that $\|z_0\|_\infty$ and $\|z_m\|_\infty$ can be clearly distinguished. We summarize our experimental results in Table 4.2 with several parameter sets. We have made the source code of our experimental programs available.[11]

4.5.5 Attack Against KR^{D}-Security of $M[\tilde{\Pi}]$ for "Artificial" Π

We construct an (artificial) IND-CPA-secure, dynamically approximately correct FHE scheme $\tilde{\Pi}$ such that $M[\tilde{\Pi}]$ fails to be KR^{D}-secure.

Theorem 20. *There exists an* IND-CPA*-secure, dynamically approximately correct FHE scheme* $\tilde{\Pi}$ *such that for any linear* ρ*-KLDP mechanism* M *that is* τ*-sensitive at 0,* $M[\tilde{\Pi}]$ *is not* KR^{D}*-secure.*

[11] https://github.com/ucsd-crypto/DynamicEstimationAttack

Proof. Let Π be any (exact) FHE scheme with message space $\mathscr{M}\mathbb{Z}_Q^n$, and assume $Q \geq n$, where n is the number of bits in the secret key $\mathsf{sk} \in \{0,1\} \in \{0,1\}^n$. Let $\mathscr{L}$ be the space of evaluatable functions. Let $\mathscr{M}' = \mathbb{Z}_Q^{n-1} \times \{0\}$, and let $\mathscr{L}' \subseteq \mathscr{L}$ be the subset of $\mathscr{L}$ that maps $\mathscr{M}' \subset \mathscr{M}$ to $\mathscr{M}'$. Define the modified decryption function

$$\mathsf{Dec}'_{\mathsf{sk}}(\mathsf{ct}) = \begin{cases} \mathsf{Dec}_{\mathsf{sk}}(\mathsf{ct}) + (1,1,\ldots,1) & \mathsf{sk}_{\mathsf{Dec}_{\mathsf{sk}}(0) \bmod n} = 0 \\ \mathsf{Dec}_{\mathsf{sk}}(\mathsf{ct}) + \tau(1,1,\ldots,1) & \mathsf{sk}_{\mathsf{Dec}_{\mathsf{sk}}(0) \bmod n} = 1 \end{cases}.$$

This is an (inexact) FHE scheme Π with message space $\mathscr{M}'$ and space of evaluatable functions $\mathscr{L}'$. This scheme is IND-CPA-secure as we have only modified the decryption algorithm (which does not impact IND-CPA security). This scheme is additionally dynamically approximately correct, as one can exactly recover the error via examining the last coordinate, and can then (exactly) compute the norm of the error. Note that as norms are homogeneous, the two possible estimates differ by the multiplicative factor τ.

We show how an adversary can recover an arbitrary bit of the key. Decryptions of $M[\tilde{\Pi}]$ are of the form $M_{T\|(1,1,\ldots,1)\|}(m')$ for $T \in \{1,\tau\}$ (depending on the value of $\mathsf{sk}_{m'[0] \bmod n}$). Subtract off m' to reduce the problem to determining the value of T from the distribution $M_{T\|(1,1,\ldots,1)\|}(0)$. As M is τ-sensitive at 0, the distributions $M_{\|(1,1,\ldots,1)\|}(0)$ and $M_{\tau\|(1,1,\ldots,1)\|}(0)$ are computationally distinguishable. One can use such a distinguisher to recover T from $M_{T\|(1,1,\ldots,1)\|}(0)$. Iterate this attack to recover the entirety of sk. □

4.6 Conclusion and Open Problems

In this work, we have shown that for CKKS with "static" error estimates, to obtain c-bits of computational IND-CPA$^{\mathsf{D}}$ security

- it suffices to add $c/2 + \tilde{O}(1)$ bits of noise (Theorem 17), and
- it is necessary to add $c/4 - \tilde{\Omega}(1)$ bits of noise (Theorem 18).

Our results therefore somewhat tightly characterize the impact on the accuracy of CKKS instantiated with a natural countermeasure to the Li-Micciancio attack [52] — $\Theta(c)$ additional bits of noise are both necessary and sufficient for security. Still, it is natural to wonder if the right scaling for our countermeasures is $c/4$ or $c/2$.

We show that our countermeasure behaves better with respect to (c,s)-bit security. In particular, we show that $s/2+\tilde{O}(1)$ bits of additional noise suffice to achieve (c,s)-bits of q-IND-CPA$^{\mathsf{D}}$ security, where s can plausibly be set much less than 128.

We include discussion of the concrete overhead of our countermeasure in Section 4.4.5, where find that our countermeasure is easily implementable (for general purpose computation) provided the FHE library supports 128-bit precision computations, while FHE libraries that support 64-bit precision computations may only be able to instantiate our countermeasure for certain (low-precision) applications, or with aggressive parameterizations.

Both our work and the work of [52] investigate how the *correctness* of encryption can impact the underlying *security* one attains. As correctness analysis typically leverages (unproven) heuristics for tighter noise estimates, we view formally justifying these heuristics to be important going forward, as the false heuristics may lead to security issues.

While our results on "dynamic" error estimation are negative, we have not ruled out achieving some weaker security notion with these techniques (for natural schemes). Our attack of Theorem 19 shows that dynamic error estimation can leak the norm of the input to the computation. Can the leakage be *provably* limited to this information?

Finally, our work examines *black box* modifications one can make to CKKS to attain q-IND-CPA$^{\mathsf{D}}$-security. It is plausible that a CKKS-specific construction could attain smaller parameters, say by randomizing homomorphic operations, choosing larger than typical scaling factors Δ, or carefully investigating the plaintext error after bootstrapping.

4.7 Acknowledgments

Chapter 4, in full, is a reprint of the material as it appears in Advances in Cryptology — CRYPTO 2022. Li, Baiyu; Micciancio, Daniele; Schultz-Wu, Mark; Sorrell, Jessica. "Securing Approximate Homomorphic Encryption Using Differential Privacy". The dissertation author was a primary investigator and author of this paper.

Bibliography

[1] Parhat Abla, Feng-Hao Liu, Han Wang, and Zhedong Wang. Ring-based identity based encryption - asymptotically shorter MPK and tighter security. In Kobbi Nissim and Brent Waters, editors, *TCC 2021: 19th Theory of Cryptography Conference, Part III*, volume 13044 of *Lecture Notes in Computer Science*, pages 157–187, Raleigh, NC, USA, November 8–11, 2021. Springer, Heidelberg, Germany.

[2] Martin Albrecht, Melissa Chase, Hao Chen, Jintai Ding, Shafi Goldwasser, Sergey Gorbunov, Shai Halevi, Jeffrey Hoffstein, Kim Laine, Kristin Lauter, Satya Lokam, Daniele Micciancio, Dustin Moody, Travis Morrison, Amit Sahai, and Vinod Vaikuntanathan. Homomorphic encryption security standard. Technical report, HomomorphicEncryption.org, Toronto, Canada, November 2018.

[3] Martin R. Albrecht, Lorenzo Grassi, Christian Rechberger, Arnab Roy, and Tyge Tiessen. MiMC: Efficient encryption and cryptographic hashing with minimal multiplicative complexity. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *Advances in Cryptology – ASIACRYPT 2016, Part I*, volume 10031 of *Lecture Notes in Computer Science*, pages 191–219, Hanoi, Vietnam, December 4–8, 2016. Springer, Heidelberg, Germany.

[4] Tomer Ashur, Mohammad Mahzoun, and Dilara Toprakhisar. Chaghri - A FHE-friendly block cipher. In Heng Yin, Angelos Stavrou, Cas Cremers, and Elaine Shi, editors, *ACM CCS 2022: 29th Conference on Computer and Communications Security*, pages 139–150, Los Angeles, CA, USA, November 7–11, 2022. ACM Press.

[5] László Babai. On lovász'lattice reduction and the nearest lattice point problem. *Combinatorica*, 6(1):1–13, 1986.

[6] Shi Bai, Tancrède Lepoint, Adeline Roux-Langlois, Amin Sakzad, Damien Stehlé, and Ron Steinfeld. Improved security proofs in lattice-based cryptography: Using the Rényi divergence rather than the statistical distance. *J. Cryptology*, 31(2):610–640, 2018.

[7] Abhishek Banerjee, Chris Peikert, and Alon Rosen. Pseudorandom functions and lattices. In Pointcheval and Johansson [70], pages 719–737.

[8] Daniel J. Bernstein and Tanja Lange. Non-uniform cracks in the concrete: The power of free precomputation. In Kazue Sako and Palash Sarkar, editors, *Advances in Cryptology – ASIACRYPT 2013, Part II*, volume 8270 of *Lecture Notes in Computer Science*, pages 321–340, Bengalore, India, December 1–5, 2013. Springer, Heidelberg, Germany.

[9] Zvika Brakerski, Nico Döttling, Sanjam Garg, and Giulio Malavolta. Leveraging linear decryption: Rate-1 fully-homomorphic encryption and time-lock puzzles. In Hofheinz and Rosen [41], pages 407–437.

[10] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (Leveled) fully homomorphic encryption without bootstrapping. In Shafi Goldwasser, editor, *ITCS 2012: 3rd Innovations in Theoretical Computer Science*, pages 309–325, Cambridge, MA, USA, January 8–10, 2012. Association for Computing Machinery.

[11] Zvika Brakerski and Vinod Vaikuntanathan. Efficient fully homomorphic encryption from (standard) LWE. In Rafail Ostrovsky, editor, *52nd Annual Symposium on Foundations of Computer Science*, pages 97–106, Palm Springs, CA, USA, October 22–25, 2011. IEEE Computer Society Press.

[12] GJ Butler. Simultaneous packing and covering in euclidean space. *Proceedings of the London Mathematical Society*, 3(4):721–735, 1972.

[13] Clément L Canonne. A survey on distribution testing: Your data is big. but is it blue? *Theory of Computing*, pages 1–100, 2020.

[14] Clément L Canonne, Gautam Kamath, and Thomas Steinke. The discrete gaussian for differential privacy. *Advances in Neural Information Processing Systems*, 33:15676–15688, 2020.

[15] Anthony Carbery and James Wright. Distributional and l-q norm inequalities for polynomials over convex bodies in r-n. *Mathematical Research Letters*, 8:233–248, 2001.

[16] Jung Hee Cheon, Kyoohyung Han, Seong-Min Hong, Hyoun Jin Kim, Junsoo Kim, Suseong Kim, Hosung Seo, Hyungbo Shim, and Yongsoo Song. Toward a secure drone system: Flying with real-time homomorphic authenticated encryption. *IEEE Access*, 6:24325–24339, 2018.

[17] Jung Hee Cheon, Kyoohyung Han, Andrey Kim, Miran Kim, and Yongsoo Song. Bootstrapping for approximate homomorphic encryption. In *EUROCRYPT 2018, Part I*, volume 10820 of *LNCS*, pages 360–384. Springer, 2018.

[18] Jung Hee Cheon, Kyoohyung Han, Andrey Kim, Miran Kim, and Yongsoo Song. A full RNS variant of approximate homomorphic encryption. In *SAC 2018*, volume 11349 of *LNCS*, pages 347–368. Springer, 2018.

[19] Jung Hee Cheon, Andrey Kim, Miran Kim, and Yong Soo Song. Homomorphic encryption for arithmetic of approximate numbers. In Tsuyoshi Takagi and Thomas Peyrin, editors, *Advances in Cryptology – ASIACRYPT 2017, Part I*, volume 10624 of *Lecture Notes in Computer Science*, pages 409–437, Hong Kong, China, December 3–7, 2017. Springer, Heidelberg, Germany.

[20] Jung Hee Cheon, Andrey Kim, Miran Kim, and Yong Soo Song. Homomorphic encryption for arithmetic of approximate numbers. In *ASIACRYPT 2017, Part I*, volume 10624 of *LNCS*, pages 409–437. Springer, 2017.

[21] Jung Hee Cheon, Andrey Kim, and Donggeon Yhee. Multi-dimensional packing for HEAAN for approximate matrix arithmetics. *IACR Cryptology ePrint Archive*, 2018:1245, 2018.

[22] Jung Hee Cheon, Duhyeong Kim, Yongdai Kim, and Yongsoo Song. Ensemble method for privacy-preserving logistic regression based on homomorphic encryption. *IEEE Access*, 6:46938–46948, 2018.

[23] J. H. Conway and N. J. A. Sloane. *Sphere Packings, Lattices and Groups*, volume 290 of *Grundlehren der mathematischen Wissenschaften*. Springer New York, New York, NY, 1999.

[24] Jan-Pieter D'Anvers, Angshuman Karmakar, Sujoy Sinha Roy, and Frederik Vercauteren. Saber: Module-LWR based key exchange, CPA-secure encryption and CCA-secure KEM. In Antoine Joux, Abderrahmane Nitaj, and Tajjeeddine Rachidi, editors, *AFRICACRYPT 18: 10th International Conference on Cryptology in Africa*, volume 10831 of *Lecture Notes in Computer Science*, pages 282–305, Marrakesh, Morocco, May 7–9, 2018. Springer, Heidelberg, Germany.

[25] H. Davenport. The covering of space by spheres. *Rendiconti del Circolo Matematico di Palermo*, 1(1):92–107, January 1952.

[26] Anindya De, Luca Trevisan, and Madhur Tulsiani. Time space tradeoffs for attacks against one-way functions and PRGs. In Tal Rabin, editor, *Advances in Cryptology – CRYPTO 2010*, volume 6223 of *Lecture Notes in Computer Science*, pages 649–665, Santa Barbara, CA, USA, August 15–19, 2010. Springer, Heidelberg, Germany.

[27] Luc Devroye, Abbas Mehrabian, and Tommy Reddad. The total variation distance between high-dimensional Gaussians with the same mean. arXiv preprint arXiv:1810.08693, 2018. https://arxiv.org/abs/1810.08693.

[28] Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, Peter Schwabe, Gregor Seiler, and Damien Stehlé. CRYSTALS-Dilithium: A lattice-based digital signature scheme. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2018(1):238–268, 2018. https://tches.iacr.org/index.php/TCHES/article/view/839.

[29] Léo Ducas and Wessel PJ van Woerden. The closest vector problem in tensored root lattices of type a and in their duals. *Designs, Codes and Cryptography*, 86:137–150, 2018.

[30] Robert E Gaunt. The basic distributional theory for the product of zero mean correlated normal random variables. *Statistica Neerlandica*, 2022.

[31] Nicholas Genise, Daniele Micciancio, Chris Peikert, and Michael Walter. Improved discrete gaussian and subgaussian analysis for lattice cryptography. In Aggelos Kiayias, Markulf Kohlweiss, Petros Wallden, and Vassilis Zikas, editors, *PKC 2020: 23rd International Conference on Theory and Practice of Public Key Cryptography, Part I*, volume 12110 of *Lecture Notes in Computer Science*, pages 623–651, Edinburgh, UK, May 4–7, 2020. Springer, Heidelberg, Germany.

[32] Nicholas Genise, Daniele Micciancio, and Yuriy Polyakov. Building an efficient lattice gadget toolkit: Subgaussian sampling and more. In Yuval Ishai and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2019, Part II*, volume 11477 of *Lecture Notes in Computer Science*, pages 655–684, Darmstadt, Germany, May 19–23, 2019. Springer, Heidelberg, Germany.

[33] Craig Gentry and Shai Halevi. Compressible FHE with applications to PIR. In Hofheinz and Rosen [41], pages 438–464.

[34] Craig Gentry, Shai Halevi, and Nigel P. Smart. Homomorphic evaluation of the AES circuit. In *CRYPTO 2012*, volume 7417 of *LNCS*, pages 850–867. Springer, 2012.

[35] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *J. Comput. Syst. Sci.*, 28(2):270–299, 1984.

[36] Naira Grigoryan, Ashot Harutyunyan, Svyatoslav Voloshynovskiy, and Oleksiy Koval. On multiple hypothesis testing with rejection option. In *2011 IEEE Information Theory Workshop*, pages 75–79. IEEE, 2011.

[37] Siyao Guo, Pritish Kamath, Alon Rosen, and Katerina Sotiraki. Limits on the efficiency of (ring) LWE-based non-interactive key exchange. *Journal of Cryptology*, 35(1):1, January 2022.

[38] Kyoohyung Han, Seungwan Hong, Jung Hee Cheon, and Daejun Park. Logistic regression on homomorphic encrypted data at scale. In *AAAI 2019*, pages 9466–9471. AAAI Press, 2019.

[39] HElib (release 2.2.0). https://github.com/homenc/HElib, 2021. IBM.

[40] Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz. A modular analysis of the Fujisaki-Okamoto transformation. In Yael Kalai and Leonid Reyzin, editors, *TCC 2017: 15th Theory of Cryptography Conference, Part I*, volume 10677 of *Lecture Notes in Computer Science*, pages 341–371, Baltimore, MD, USA, November 12–15, 2017. Springer, Heidelberg, Germany.

[41] Dennis Hofheinz and Alon Rosen, editors. *TCC 2019: 17th Theory of Cryptography Conference, Part II*, volume 11892 of *Lecture Notes in Computer Science*, Nuremberg, Germany, December 1–5, 2019. Springer, Heidelberg, Germany.

[42] Jiantao Jiao, Thomas Courtade, Albert No, Kartik Venkat, and Tsachy Weissman. Information divergences and the curious case of the binary alphabet. In *2014 IEEE International Symposium on Information Theory*, pages 351–355. IEEE, 2014.

[43] Zhengzhong Jin and Yunlei Zhao. Generic and practical key establishment from lattice. In Robert H. Deng, Valérie Gauthier-Umaña, Martín Ochoa, and Moti Yung, editors, *ACNS 19: 17th International Conference on Applied Cryptography and Network Security*, volume 11464 of *Lecture Notes in Computer Science*, pages 302–322, Bogota, Colombia, June 5–7, 2019. Springer, Heidelberg, Germany.

[44] Dhiraj D. Kalamkar, Dheevatsa Mudigere, Naveen Mellempudi, Dipankar Das, Kunal Banerjee, Sasikanth Avancha, Dharma Teja Vooturi, Nataraj Jammalamadaka, Jianyu Huang, Hector Yuen, Jiyan Yang, Jongsoo Park, Alexander Heinecke, Evangelos Georganas, Sudarshan Srinivasan, Abhisek Kundu, Misha Smelyanskiy, Bharat Kaul, and Pradeep Dubey. A study of BFLOAT16 for deep learning training. arXiv preprint arXiv:1905.12322, 2019. https://arxiv.org/abs/1905.12322.

[45] Andrey Kim, Antonis Papadimitriou, and Yuriy Polyakov. Approximate homomorphic encryption with reduced approximation error. In *CT-RSA*, volume 13161 of *Lecture Notes in Computer Science*, pages 120–144. Springer, 2022.

[46] Bo'az Klartag. Logarithmic bounds for isoperimetry and slices of convex sets, 2023.

[47] Anusha Lalitha and Tara Javidi. On error exponents of almost-fixed-length channel codes and hypothesis tests. *arXiv preprint arXiv:2012.00077*, 2020.

[48] Lattigo 2.2.0. Online: http://github.com/ldsec/lattigo, July 2021. EPFL-LDS.

[49] Keewoo Lee. Bit security as cost to observe advantage: Towards the definition from the book. *Cryptology ePrint Archive*, 2022.

[50] Yin Tat Lee and Santosh S Vempala. The kannan–lovász–simonovits conjecture. *Current Developments in Mathematics*, 2017(1):1–36, 2017.

[51] Leonid A. Levin. Randomness and non-determinism. European Summer Meeting of the Association for Symbolic Logic, 1993.

[52] Baiyu Li and Daniele Micciancio. On the security of homomorphic encryption on approximate numbers. In Anne Canteaut and François-Xavier Standaert, editors, *Advances in Cryptology – EUROCRYPT 2021*, pages 648–677, Cham, 2021. Springer International Publishing.

[53] Baiyu Li, Daniele Micciancio, Mark Schultz, and Jessica Sorrell. Securing approximate homomorphic encryption using differential privacy. In Yevgeniy Dodis and Thomas Shrimpton, editors, *Advances in Cryptology – CRYPTO 2022, Part I*, volume 13507 of *Lecture Notes in Computer Science*, pages 560–589, Santa Barbara, CA, USA, August 15–18, 2022. Springer, Heidelberg, Germany.

[54] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. *J. ACM*, 60(6):43:1–43:35, 2013.

[55] Jacques Martinet. *Perfect Lattices in Euclidean Spaces*, volume 327 of *Grundlehren der mathematischen Wissenschaften*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003.

[56] Robby G. McKilliam, Warren D. Smith, and I. Vaughan L. Clarkson. Linear-Time Nearest Point Algorithms for Coxeter Lattices. *IEEE Transactions on Information Theory*, 56(3):1015–1022, March 2010.

[57] Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In Pointcheval and Johansson [70], pages 700–718.

[58] Daniele Micciancio and Yuriy Polyakov. Bootstrapping in fhew-like cryptosystems. In *Proceedings of the 9th on Workshop on Encrypted Computing & Applied Homomorphic Cryptography*, pages 17–28, 2021.

[59] Daniele Micciancio and Michael Walter. Gaussian sampling over the integers: Efficient, generic, constant-time. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology – CRYPTO 2017, Part II*, volume 10402 of *Lecture Notes in Computer Science*, pages 455–485, Santa Barbara, CA, USA, August 20–24, 2017. Springer, Heidelberg, Germany.

[60] Daniele Micciancio and Michael Walter. Gaussian sampling over the integers: Efficient, generic, constant-time. In *CRYPTO (2)*, volume 10402 of *Lecture Notes in Computer Science*, pages 455–485. Springer, 2017.

[61] Daniele Micciancio and Michael Walter. On the bit security of cryptographic primitives. In Jesper Buus Nielsen and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2018, Part I*, volume 10820 of *Lecture Notes in Computer Science*, pages 3–28, Tel Aviv, Israel, April 29 – May 3, 2018. Springer, Heidelberg, Germany.

[62] Daniele Micciancio and Michael Walter. On the bit security of cryptographic primitives. In *EUROCRYPT (1)*, volume 10820 of *Lecture Notes in Computer Science*, pages 3–28. Springer, 2018.

[63] Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275, 2017.

[64] PALISADE lattice cryptography library (release 1.11.6). https://gitlab.com/palisade/, 2022. PALISADE Project.

[65] Saerom Park, Jaewook Lee, Jung Hee Cheon, Juhee Lee, Jaeyun Kim, and Junyoung Byun. Security-preserving support vector machine with fully homomorphic encryption. In *SafeAI@AAAI 2019*, volume 2301 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2019.

[66] Chris Peikert. Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In Michael Mitzenmacher, editor, *41st Annual ACM Symposium on Theory of Computing*, pages 333–342, Bethesda, MD, USA, May 31 – June 2, 2009. ACM Press.

[67] Chris Peikert. Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In *STOC*, pages 333–342. ACM, 2009.

[68] Chris Peikert, Oded Regev, and Noah Stephens-Davidowitz. Pseudorandomness of ring-lwe for any ring and modulus. In *STOC*, pages 461–473. ACM, 2017.

[69] Ankit Pensia, Varun Jog, and Po-Ling Loh. Communication-constrained hypothesis testing: Optimality, robustness, and reverse data processing inequalities. *IEEE Transactions on Information Theory*, 2023.

[70] David Pointcheval and Thomas Johansson, editors. *Advances in Cryptology – EUROCRYPT 2012*, volume 7237 of *Lecture Notes in Computer Science*, Cambridge, UK, April 15–19, 2012. Springer, Heidelberg, Germany.

[71] Yuriy Polyakov. personal communication, October 2020.

[72] Yury Polyanskiy and Yihong Wu. Lecture notes on information theory. *Lecture Notes for ECE563 (UIUC) and*, 6(2012-2016):7, 2014.

[73] Yury Polyanskiy and Yihong Wu. Information theory: From coding to learning. *Book draft*, 2022.

[74] A.H. van Poppelen. Cryptographic Decoding of the Leech Lattice. Master's thesis, Utrecht University, 2016. https://studenttheses.uu.nl/handle/20.500.12932/24606.

[75] Thomas Pöppelmann, Léo Ducas, and Tim Güneysu. Enhanced lattice-based signatures on reconfigurable hardware. In Lejla Batina and Matthew Robshaw, editors, *Cryptographic Hardware and Embedded Systems – CHES 2014*, pages 353–370, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg.

[76] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In Harold N. Gabow and Ronald Fagin, editors, *37th Annual ACM Symposium on Theory of Computing*, pages 84–93, Baltimore, MA, USA, May 22–24, 2005. ACM Press.

[77] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6):34:1–34:40, 2009.

[78] Charbel Saliba, Laura Luzzi, and Cong Ling. A reconciliation approach to key generation based on module-lwe. In *2021 IEEE International Symposium on Information Theory (ISIT)*, pages 1636–1641, 2021.

[79] Igal Sason. On reverse pinsker inequalities. *arXiv preprint arXiv:1503.07118*, 2015.

[80] Adrien Saumard and Jon A Wellner. Log-concavity and strong log-concavity: a review. *Statistics surveys*, 8:45, 2014.

[81] Microsoft SEAL (release 3.6). https://github.com/Microsoft/SEAL, November 2020. Microsoft Research, Redmond, WA.

[82] Ananda Theertha Suresh. Robust hypothesis testing and distribution estimation in hellinger distance. In *International Conference on Artificial Intelligence and Statistics*, pages 2962–2970. PMLR, 2021.

[83] Naigang Wang, Jungwook Choi, Daniel Brand, Chia-Yu Chen, and Kailash Gopalakrishnan. Training deep neural networks with 8-bit floating point numbers. *Advances in neural information processing systems*, 31, 2018.

[84] Shun Watanabe and Kenji Yasunaga. Bit security as computational cost for winning games with high probability. In Mehdi Tibouchi and Huaxiong Wang, editors, *Advances in Cryptology – ASIACRYPT 2021, Part III*, volume 13092 of *Lecture Notes in Computer Science*, pages 161–188, Singapore, December 6–10, 2021. Springer, Heidelberg, Germany.

[85] Shun Watanabe and Kenji Yasunaga. Unified view for notions of bit security. Cryptology ePrint Archive, Paper 2022/693, 2022. https://eprint.iacr.org/2022/693.

[86] Kenji Yasunaga. Replacing probability distributions in security games via hellinger distance. In *2nd Conference on Information-Theoretic Cryptography (ITC 2021)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2021.

[87] Ram Zamir, Bobak Nazer, Yuval Kochman, and Ilai Bistritz. *Lattice Coding for Signals and Networks: A Structured Coding Approach to Quantization, Modulation and Multiuser Information Theory*. Cambridge University Press, 2014.