GGHLite: More Efficient Multilinear Maps from Ideal Lattices*

Adeline Langlois[1], Damien Stehlé[1], Ron Steinfeld[2]

[1] ENS de Lyon, Laboratoire LIP (U. Lyon, CNRS, ENS de Lyon, INRIA, UCBL),
46 Allée d'Italie, 69364 Lyon Cedex 07, France.
[2] Clayton School of Information Technology, Monash University, Clayton, Australia.

Abstract. The GGH Graded Encoding Scheme [10], based on ideal lattices, is the first plausible approximation to a cryptographic multilinear map. Unfortunately, using the security analysis in [10], the scheme requires very large parameters to provide security for its underlying "encoding re-randomization" process. Our main contributions are to formalize, simplify and improve the efficiency and the security analysis of the re-randomization process in the GGH construction. This results in a new construction that we call GGHLite. In particular, we first lower the size of a standard deviation parameter of the re-randomization process of [10] from exponential to polynomial in the security parameter. This first improvement is obtained via a finer security analysis of the "drowning" step of re-randomization, in which we apply the *Rényi divergence* instead of the conventional *statistical distance* as a measure of distance between distributions. Our second improvement is to reduce the number of randomizers needed from $\Omega(n \log n)$ to 2, where n is the dimension of the underlying ideal lattices. These two contributions allow us to decrease the bit size of the public parameters from $O(\lambda^5 \log \lambda)$ for the GGH scheme to $O(\lambda \log^2 \lambda)$ in GGHLite, with respect to the security parameter λ (for a constant multilinearity parameter κ).

1 Introduction

Boneh and Silverberg [7] defined a *cryptographic κ-multilinear map* e as a map from $G_1 \times \ldots \times G_\kappa$ to G_T, all cyclic groups of order p, which enjoys three main properties: first, for any elements $g_i \in G_i$ for $i \leq \kappa$, $j \leq \kappa$ and $\alpha \in \mathbb{Z}_p$, we have $e(g_1, \ldots, \alpha \cdot g_j, \ldots, g_\kappa) = \alpha \cdot e(g_1, \ldots, g_\kappa)$; second, the map e is non-degenerate, i.e., if the g_i's are generators of their respective G_i's then $e(g_1, \ldots, g_\kappa)$ generates G_T; and third, there is no efficient algorithm to compute discrete logarithms in any of the G_i's. Bilinear maps ($\kappa = 2$) and multilinear maps have a lot of cryptographic applications, see [16,27,6] and [7,26,22,25], respectively. But unlike bilinear maps, built with pairings on elliptic curves, the construction of cryptographic multilinear maps was an open problem for several years. In [7], Boneh and Silverberg studied the interest of such maps, and gave two applications: multipartite Diffie-Hellman key exchange and very efficient broadcast encryption. But they conjectured that multilinear maps will probably "come from outside the realm of algebraic geometry." In 2013, Garg, Gentry and Halevi [10] introduced the first "approximate" multilinear maps contruction, based on ideal lattices, and the powerful notion of *graded encoding scheme*. Based on their work, Coron, Lepoint and Tibouchi [8] recently described an alternative construction of graded encoding scheme.

We first give a high level description of the GGH graded encoding scheme [10]. If we come back to the definition of cryptographic multilinear maps, the authors of [10] notice that $\alpha \cdot g_i$ can be viewed as an "encoding" of the "plaintext" $\alpha \in \mathbb{Z}_q$. They consider the polynomial rings $R = \mathbb{Z}[x]/\langle x^n + 1 \rangle$ and $R_q = R/qR$ (replacing the exponent space $\mathbb{Z}_p$). They generate a small secret $g \in R$ and let $\mathcal{I} = \langle g \rangle$ be the principal ideal over R generated by g. They also sample a uniform $z \in R_q$ which stays secret. The "plaintext" is an element of $R/\mathcal{I}$, and is encoded via a division by z in R_q: to

* This is the full version of a paper presented at the EUROCRYPT 2014 conference.

encode a coset of $R/\mathcal{I}$, return $[c/z]_q$, where c is an arbitrary small coset representative. In practice, as g is hidden, they give another public parameter y, which is an encoding of 1, and the encoding of the coset is computed as $[e \cdot y]_q$, where e is a small coset representative (possibly different from c). But, as opposed to multilinear maps, their graded encoding scheme uses the notion of *encoding level*: the plaintext e is a level-0 encoding, the encoding $[c/z]_q$ is a level-1 encoding, and at level i, an encoding of $e + \mathcal{I}$ is given by $[c/z^i]_q = [e \cdot y^i]_q$. These encodings are both additively and multiplicatively homomorphic, up to a limited number of operations. More precisely, a product of i level-1 encodings is a level-i encoding. One can multiply any number of encodings up to κ, instead of exactly κ in multilinear maps (the parameter κ is called the multilinearity parameter).

The authors of [10] introduced new hardness assumptions: the Graded Decisional Diffie-Hellman (GDDH) and its computational variant (GCDH). These are natural analogues of the Diffie-Hellman problems from group-based cryptography. To ensure their hardness, and hence the security of the cryptographic constructions, the second main difference with multilinear maps is the randomization of the encodings. The principle is as follows: first some level-1 encodings of 0, called $\{x_j = [b_j/z]_q\}_{j \leq m_r}$, are given as part of the public parameters; then, to randomize a level-1 encoding $u' = [e \cdot y]_q$, one outputs $u = [u' + \sum_j \rho_j x_j]_q = [c/z]_q$ with $c = c' + \sum_j \rho_j b_j$, where the ρ_j's are sampled from a discrete Gaussian distribution over $\mathbb{Z}$ with deviation parameter σ^*. Without this re-randomization, the encoding u' of e allows e to be efficiently recovered using $u = [u'y^{-1}]_q$. Adding the re-randomization step prevents this division attack, but the statistical properties of the distribution of the re-randomized encoding u remain correlated to some extent with the original encoding u' (for instance, the center of the distribution of c is c', since the distribution of $\sum_j \rho_j b_j$ is known to be centered at 0). This property may allow other attacks that exploit this correlation. The question arises as to how to set the re-randomization parameter σ^* in order to guarantee security against such potential "statistical correlation" attacks – the larger the re-randomization parameters the smaller the correlation, and heuristically the more resistant the scheme is to such attacks. But increasing σ^* impacts the efficiency of the scheme.

In [10], the authors use a "drowning step" to solve this problem. This technique, also called "smudging," was previously used in other applications [4,13,3,5]. Generally, "drowning" consists in hiding a secret vector $\boldsymbol{s} \in \mathbb{Z}^n$ by adding a sufficiently large random noise $\boldsymbol{e} \in \mathbb{Z}^n$ to it, so that the distribution of $\boldsymbol{s} + \boldsymbol{e}$ becomes "almost independent" of $\boldsymbol{s}$. In all of the above applications, to achieve a security level 2^λ (where λ denotes the security parameter), the security analysis requires "almost independent" to be interpreted as "within statistical distance $2^{-\lambda}$ from a distribution that is independent of $\boldsymbol{s}$." In turn, this requirement implies the need for "exponential drowning," i.e., the ratio $\gamma = \|\boldsymbol{e}\|/\|\boldsymbol{s}\|$ between the magnitude of the noise and the magnitude of secret needs to be $2^{\Omega(\lambda)}$. Exponential drowning imposes a severe penalty on the efficiency of these schemes, as their security is related to γ-approximation lattice problems, whose complexity decreases exponentially with $\log \gamma$. As a result, the schemes require a lattice dimension n at least quadratic in λ and key length at least cubic in λ. In summary, the GGH re-randomization step, necessary for its security, is also a primary factor in its inefficiency.

Our contributions. First, we formalize the re-randomization security goal in the GGH construction, that is implicit in the work of [10]. A primary security goal of re-randomization is to guarantee security of the GDDH problem against statistical correlation attacks. Accordingly, we formulate a security goal that captures this security guarantee, by introducing a canonical variant of GDDH, called cGDDH. In this variant, the encodings of some elements are sampled from a canonical distribution whose statistical properties are independent of the encoded elements. Consequently, the

canonical problems are by construction not subject to "statistical correlation" attacks. Our re-randomization security goal is formulated as the existence of an efficient computational reduction from the canonical problems to their corresponding non-canonical variants.

Our first main improvement to the GGH scheme relies on a new security analysis of the drowning step in the GGH re-randomization algorithm. We show that our re-randomization security goal can be satisfied *without* "exponential drowning," thus removing the main efficiency bottleneck. Namely, our analysis provides a re-randomization at security level 2^λ while allowing the use of a re-randomization deviation parameter σ^* that only drowns the norm of the randomness offset $r' \in \mathcal{I}$ (from the original encoding to be re-randomized) by a *polynomial* (or even constant) drowning ratio $\gamma = \lambda^{O(1)}$ (rather than $\gamma = 2^{\Omega(\lambda)}$, as needed in the analysis of [10]). However, our analysis only works for the search variant of the Graded Diffie-Hellman problem. Fortunately, we show that the two flagship applications of the GGH scheme – the N-party Key Agreement [10] and the Attribute Based Encryption [12] – can be modified to rely on this computational assumption (in the random oracle model).

Our second main improvement of the re-randomization process is to decrease m_r, the number of encodings of 0 needed, from $\Omega(n \log n)$ to 2. We achieve this result by presenting a new discrete Gaussian Leftover Hash Lemma (LHL) over algebraic rings. In [10], the authors apply the discrete Gaussian LHL from [2] to show that the distribution of the sum $\sum_{j \le m_r} \rho_j r_j$ is close to a discrete Gaussian on the ideal $\mathcal{I}$. Our improvement consists in sampling the randomizers ρ_j as elements of the full n-dimensional ring R, rather than just from $\mathbb{Z}$. Since each randomizer now has n times more entropy than before, one may hope to obtain a similar LHL result as in [2] while reducing m_r by a factor $\approx n$. However, as the designers of the GGH scheme notice in [10, Se. 6.4], the proof techniques from [2] do not seem to immediately carry over to our "algebraic ring" LHL setting. Our new LHL over rings resolves this problem.

The two contributions above allow us to decrease the bit size of the public parameters from $O(\kappa^3 \lambda^5 \log(\kappa\lambda))$ for the GGH scheme to $O(\kappa^3 \lambda \log^2(\kappa\lambda))$ for GGHLite, for multilinearity factor κ and security level 2^λ for the graded Diffie-Hellman problem.

Technical overview. Our first main result is to reduce the size of the parameter σ^* in the re-randomization process. Technically, our improved analysis of drowning is obtained by using the *Rényi divergence* (RD) to replace the conventional statistical distance (SD) as a measure of distribution closeness. The RD was already exploited in a different context in [18, Claim 5.11], to show the hardness of Ring-LWE. Here, we use the RD to decrease the amount of drowning, by bounding the RD between a discrete Gaussian distribution and its offset. This suffices for relating the hardness of the search problems using these encoding distributions, even though the SD between the distributions is non-negligible. The technique does not seem to easily extend to the decision problems, as RD induces a multiplicative relationship between success probabilities, rather than an additive relationship as SD does.

Our second main result is a new LHL over the ring R. We now briefly explain this result and its proof. For a fixed $X = [x_1, x_2] \in R^2$, with each x_i sampled from $D_{R,s}$, our goal is to study the distribution $\widetilde{\mathcal{E}}_{X,s} = x_1 \cdot D_{R,s} + x_2 \cdot D_{R,s}$. In particular, we prove that $\widetilde{\mathcal{E}}_{X,s}$ is statistically close to $D_{\mathbb{Z}^n, sX^T}$. For this, we adapt the proof of the LHL in [2]: we follow a similar series of steps, but the proofs of these steps differ technically, as we exploit the ring structure.

We first show that $X \cdot R^2 = R$, except with some constant probability < 1. For this, we adapt a result from [29] on the probability that two Gaussian samples of R are coprime. Note that in contrast to the LHL over $\mathbb{Z}$ in [2], in our setting the probability that $X \cdot R^2 \neq R$ is non-negligible.

This is unavoidable with the ring $R = \mathbb{Z}[x]/\langle x^n + 1\rangle$, since each random element of R falls in the ideal $\langle x+1\rangle$ with probability $\approx 1/2$, both x_1 and x_2 (and hence the ideal they generate) get "stuck" in $\langle x+1\rangle$ with probability $\approx 1/4$. However, the probability of this bad event is bounded away from 1 by a constant and thus we only need a constant number of trials on average with random X's to obtain a good X by rejection.

Then, we define the orthogonal R-module $A_X = \{\boldsymbol{v} \in R^2 : X \cdot \boldsymbol{v} = 0\}$, and apply a directly adapted variant of [2, Le. 10] to show that if the parameter s is larger than the smoothing parameter $\eta_\varepsilon(A_X)$ (with A_X viewed as an integral lattice), then the SD between $\widetilde{\mathcal{E}}_{X,s}$ and the ellipsoidal Gaussian $D_{\mathbb{Z}^n, sX^T}$ is bounded by 2ε. We finally show that this condition on the smoothing parameter of A_X holds. For this, we observe that the Minkowski minima of the lattice A_X are equal, due to the R-module structure of A_X. This allows us to bound the last minimum from above using Minkowski's second theorem. A similar approach was previously used (e.g., in [17]) to bound the smoothing parameter of ideal lattices.

Open problems. Our "Rényi divergence" technique used to make the drowning efficient in the GGH scheme is likely to have many applications. It is an interesting open problem to see whether it could be used to remove exponential drowning in other contexts, such as [4,13], and whether it could be used for a wider class of decision problems, such as as GDDH. We also think our new LHL over rings may have other applications.

A very important open question is to gain a better understanding of the complexity of the canonical Ext-GCDH problem and its variants, or to modify GGHLite to make its security based on more well studied problems. Our "NTRU variant" of GGHLite seems somewhat closer to the NTRUEncrypt scheme [15], and may be a first step in this direction. It is also intriguing to understand better the security connection between this construction and the jigsaw puzzle variant used in the construction of a candidate indistinguishability obfuscation mechanism [11].

Finally, evaluating the concrete computational and space efficiency achievable by GGHLite, by setting the parameters based on the best known attacks on the underlying canonical problems (and comparison with the concrete implementation of the integer-based scheme reported in [8]) is another direction for future work.

Roadmap. The rest of this paper is organized as follows. In Section 2, we summarize notation and necessary background (some additional background is postponed to the appendices). Section 3 reviews the GGH multilinear map construction, its underlying computational problems, and the strong re-randomization security requirement from [10], and then introduces our canonical computational problems and formulates our precise security goal for re-randomization with respect to the canonical problems. In Section 4, we study the Rényi divergence as an alternative to the statistical distance in order to improve the security analysis of re-randomization "drowning" step. Section 5 contains our second main improvement to the re-randomization process: the algebraic ring variant of the discrete Gaussian leftover hash lemma from [2]. In Section 6, we show how to combine the results from the previous two sections to obtain our improved construction GGHLite. Section 7 compares the asymptotic parameters of GGHLite with those of the original GGH scheme. Finally, in Section 8, we show how to adapt some applications of multilinear maps to rely on the hardness of the Ext-GCDH problem, to which our security result for GGHLite applies.

2 Preliminaries

Notation. A function $f(\lambda)$ is said negligible if it is $\lambda^{-\omega(1)}$. For an integer q, we let $\mathbb{Z}_q$ denote the ring of integers modulo q. The notation $[\cdot]_q$ means that all operations within the square brackets are performed modulo q. We choose $n \geq 4$ as a power of 2, and let K and R respectively denote the polynomial rings $\mathbb{Q}[X]/\langle x^n + 1\rangle$ and $\mathbb{Z}[X]/\langle x^n + 1\rangle$. The rings K and R are isomorphic to the cyclotomic field of order $2n$ and its ring of integers, respectively. For an integer q, we let R_q denote the ring $\mathbb{Z}_q[x]/\langle x^n + 1\rangle \simeq R/qR$. For $z \in R$ we denote by $\mathrm{MSB}_\ell(z) \in \{0,1\}^{\ell \cdot n}$ the ℓ most-significant bits of each of the n coefficients of z. Vectors are denoted in bold. For $\boldsymbol{b} \in \mathbb{R}^d$ (resp. $g \in K$), we let $\|\boldsymbol{b}\|$ (resp. $\|g\|$) denote its Euclidean norm (resp. norm of its coefficient vector). The uniform distribution on finite set E is denoted by $U(E)$. For a random variable z, we use $D(z)$ to denote the distribution of z. The statistical distance (SD) between distributions D_1 and D_2 over a countable domain E is $\Delta(D_1, D_2) = \frac{1}{2}\sum_{x\in E} |D_1(x) - D_2(x)|$. For a function f over a countable domain E, we let $f(E) = \sum_{x\in E} f(x)$. Let $X \in \mathbb{R}^{m\times n}$ be a rank-n matrix and $U_X = \{\|X\boldsymbol{u}\| : \boldsymbol{u} \in \mathbb{R}^n, \|\boldsymbol{u}\| = 1\}$. The smallest (resp. largest) singular value of X is denoted by $\sigma_n(X) = \inf(U_X)$ (resp. $\sigma_1(X) = \sup(U_X)$).

Lattices. We refer to [19,23] for introductions to the computational aspects of lattices. A d-dimensional *lattice* $\Lambda \subseteq \mathbb{R}^n$ is the set of all integer linear combinations $\sum_{i=1}^d x_i\boldsymbol{b}_i$ of some linearly independent vectors $\boldsymbol{b}_i \in \mathbb{R}^n$. The determinant $\det(\Lambda)$ is defined as $\sqrt{\det(B^TB)}$, where $B = (\boldsymbol{b}_i)_i$ is any such *basis* of Λ. For $i \leq d$, the ith minimum $\lambda_i(\Lambda)$ is the smallest r such that Λ contains i linearly independent vectors of norms $\leq r$.

Lemma 2.1 (Minkowski's second theorem). *Let Λ be an n-dimensional lattice. Then:*

$$\Big(\prod_{1\leq n} \lambda_i(\Lambda)\Big)^{1/n} \leq \sqrt{n}\det(\Lambda)^{1/n}.$$

The following result links the determinants of a lattice and its orthogonal.

Lemma 2.2 ([21, Cor. 2]). *Let $\Lambda \subseteq \mathbb{Z}^n$ be a lattice, and let $\Lambda^\perp = (\mathrm{Span}(\Lambda))^\perp \bigcap \mathbb{Z}^n$ denote the orthogonal lattice of Λ. Then* $\det(\Lambda^\perp) \leq \det(\Lambda)$.

Gaussian distributions. For a rank-n matrix $S \in \mathbb{R}^{m\times n}$ and a vector $\boldsymbol{c} \in \mathbb{R}^n$, the *ellipsoid* Gaussian distribution with parameter S and center $\boldsymbol{c}$ is defined as:

$$\forall \boldsymbol{x} \in \mathbb{R}^n, \rho_{S,\boldsymbol{c}}(x) = \exp\left(-\pi(\boldsymbol{x}-\boldsymbol{c})^T(S^TS)^{-1}(\boldsymbol{x}-\boldsymbol{c})\right).$$

Note that $\rho_{S,\boldsymbol{c}}(x) = \exp(-\pi\|(S^T)^\dagger(\boldsymbol{x}-\boldsymbol{c})\|^2)$, where $X^\dagger$ denotes the pseudo-inverse of X. The *ellipsoid* discrete Gaussian distribution over a coset $\Lambda + z$ of a lattice Λ, with parameter S and center $\boldsymbol{c}$ is defined as: $\forall \boldsymbol{x} \in \Lambda + z, D_{\Lambda+z,S,\boldsymbol{c}} = \rho_{S,\boldsymbol{c}}(\boldsymbol{x})/\rho_{S,\boldsymbol{c}}(\Lambda)$. The *truncated tail Gaussian* $D^t_{\Lambda,S,\boldsymbol{c}}$ is obtained by sampling $\boldsymbol{x}$ from $D_{\Lambda,S,\boldsymbol{c}}$, and resampling if $\|\boldsymbol{x}\| > 2\cdot\sqrt{n}\cdot\sigma_1(S)$, where n denotes the dimension of Λ. As shown in Lemma 2.3 below, the rejection probability can be made $O(2^{-n})$.

Smoothing parameter. Introduced by [20], the *smoothing parameter* $\eta_\varepsilon(\Lambda)$ of an n-dimensional lattice Λ and a real $\varepsilon > 0$ is defined as the smallest s such that $\rho_{1/s}(\Lambda^* \setminus \{0\}) \leq \varepsilon$. We use the following properties.

Lemma 2.3 ([2, Le. 3]). *For a rank-n lattice Λ, constant $0 < \varepsilon < 1$, vector $\boldsymbol{c}$ and matrix S with $\sigma_n(S) \geq \eta_\varepsilon(\Lambda)$, if $\boldsymbol{x}$ is sampled from $D_{\Lambda,S,\boldsymbol{c}}$ then $\|\boldsymbol{x}\| \leq \sigma_1(S)\sqrt{n}$, except with probability $\leq \frac{1+\varepsilon}{1-\varepsilon} \cdot 2^{-n}$.*

Lemma 2.4 ([20, Le. 3.3]). *Let Λ be an n-dimensional lattice and $\varepsilon > 0$. Then*

$$\eta_\varepsilon(\Lambda) \leq \sqrt{\frac{\ln(2n(1+1/\varepsilon))}{\pi}} \cdot \lambda_n(\Lambda).$$

Lemma 2.5 (Adapted from [20, Le. 2.7]). *Let Λ be an n-dimensional lattice and $\varepsilon \in (0,1)$. Then for any $c \in \mathbb{R}^n$ and $s \geq \eta_\varepsilon(\Lambda)$ we have $\rho_{s,c}(\Lambda) \in [1-\varepsilon, 1+\varepsilon] \cdot \det(\Lambda)^{-1}$.*

Lemma 2.6 (Adapted from [14, Cor. 2.8]). *Let Λ, Λ' be n-dimensional lattices with $\Lambda' \subseteq \Lambda$ and $\varepsilon \in (0, 1/2)$. Then for any $c \in \mathbb{R}^n$ and $s \geq \eta_\varepsilon(\Lambda')$ and any $x \in \Lambda/\Lambda'$ we have*

$$(D_{\Lambda,s,c} \bmod \Lambda')(x) \in \left[\frac{1-\varepsilon}{1+\varepsilon}, \frac{1+\varepsilon}{1-\varepsilon}\right] \cdot \frac{\det(\Lambda)}{\det(\Lambda')}.$$

Algebraic number rings and ideal lattices. For $g, x \in R$, we let $[x]_g$ denote the reduction of x modulo the principal ideal $I = \langle g \rangle$ with respect to the $\mathbb{Z}$-basis $(g, x \cdot g, \ldots, x^{n-1} \cdot g)$, i.e., $[x]_g$ is the unique element of R in $\mathcal{P}_g = \{\sum_{i=0}^{n-1} c_i x^i g : c_i \in [-1/2, 1/2) \cap \mathbb{R}\}$ such that $x - [x]_g \in \langle g \rangle$. The set $\mathcal{P}_g \cap R$ is a set of unique representatives of the cosets of I in R, that make up the quotient ring R/I.

To use our improved drowning lemma in Section 4, we need a lower bound on the least singular value $\sigma_n(\mathrm{rot}(b))$ of the matrix $\mathrm{rot}(b) \in \mathbb{Z}^{n \times n}$ corresponding to the map $x \mapsto b \cdot x$ over R, for a Gaussian distributed $b \hookleftarrow D_{I,\sigma}$. We also let $b[j] = b(\zeta^{2j+1})$ denote the jth complex embedding of b, where $\zeta \in \mathbb{C}$ is a primitive $2n$th root of unity. We define $T_2(b) = (\sum_j |b[j]|^2)^{1/2}$. Recall that we have $T_2(b)^2 = n\|b\|^2$ (see, e.g., [29]). In the proof of [29, Le. 4.1], a probabilistic lower bound on $\min_{j \in [n]} |b[j]|$ is obtained for a Gaussian distributed b. Since

$$\begin{aligned}
\sigma_n(b)^2 &= \min_{u \in K, \|u\|=1} \|u \cdot b\|^2 = \frac{1}{n} \min_{u \in K, T_2(u)^2 = n} \sum_{j \in [n]} |u[j]|^2 \cdot |b[j]|^2 \\
&= \min_{j \in [n]} |b[j]|^2 = \frac{1}{\max_{j \in [n]} |b[j]^{-1}|^2} \\
&\leq \frac{1}{\frac{1}{n}\sum_{j \in [n]} |b[j]^{-1}|^2} = \frac{1}{\|b^{-1}\|^2},
\end{aligned}$$

we can immediately adapt it to get the following.

Lemma 2.7 (Adapted from [29, Le. 4.1]). *Let $R = \mathbb{Z}^n[x]/(x^n+1)$ for n a power of 2. For any ideal $I \subseteq R$, $\delta \in (0,1)$, $t \geq \sqrt{2\pi}$ and $\sigma \geq \frac{t}{\sqrt{2\pi}} \cdot \eta_\delta(I)$, we have:*

$$\Pr_{b \hookleftarrow D_{I,\sigma}}\left[\|b^{-1}\| \geq \tfrac{t}{\sigma\sqrt{n/2}}\right] \leq \Pr_{b \hookleftarrow D_{I,\sigma}}\left[\sigma_n(b) \leq \tfrac{\sigma\sqrt{n/2}}{t}\right] \leq \tfrac{1+\delta}{1-\delta}\tfrac{n\sqrt{2\pi \mathrm{e}}}{t}.$$

We can also obtain a lower bound $\sigma_n(b)^2 \geq \frac{1}{n} \cdot \|b^{-1}\|^{-2}$ by replacing the last line in the equations above Lemma 2.7 by $\geq \frac{1}{\sum_{j \in [n]} |b[j]^{-1}|^2} = \frac{1}{n \cdot \|b^{-1}\|^2}$.

3 GGH and its re-randomization procedure

In this section, we recall the Garg et al. scheme from [10], and its related hard problems. We then discuss the re-randomization step of the scheme and explain what should be expected from it, in terms of security. This security requirement is unclear in [10] and [2]. We formulate it precisely. This will drive our re-randomization design in the following sections.

3.1 The GGH scheme

We recall the GGH scheme in Figure 1. We present it here in a slightly more general form than [10]: we leave as a parameter the distribution χ_k of the re-randomization coefficients ρ_j for a level-k encoding (for any $k \leq \kappa$). In the original GGH scheme, we have $\chi_k = D_{\mathbb{Z},\sigma_k^*}$ for some σ_k^*'s, i.e., the ρ_j's are integers sampled from a discrete Gaussian distribution. Looking ahead, in Section 5, we analyze a more efficient variant, in which $\chi_k = D_{R,\sigma_k^*}$, so that the ρ_j's belong to R.

- **Instance generation** $\mathsf{InstGen}(1^\lambda, 1^\kappa)$: Given security parameter λ and multilinearity parameter κ, determine scheme parameters n, q, m_r, σ, σ', $\ell_{g^{-1}}$, ℓ, based on the scheme analysis. Then proceed as follows:
  - Sample $g \hookleftarrow D_{R,\sigma}$ until $\|g^{-1}\| \leq \ell_{g^{-1}}$ and $\mathcal{I} = \langle g \rangle$ is a prime ideal. Define encoding domain $R_g = R/\langle g \rangle$.
  - Sample $z \hookleftarrow U(R_q)$.
  - Sample a level-1 encoding of 1: set $y = [a \cdot z^{-1}]_q$ with $a \hookleftarrow D_{1+\mathcal{I},\sigma'}$.
  - For $k \leq \kappa$, sample m_r level-k encodings of 0: set $x_j^{(k)} = [b_j^{(k)} \cdot z^{-k}]_q$ with $b_j^{(k)} \hookleftarrow D_{\mathcal{I},\sigma'}$ for all $j \leq m_r$.
    (Note that $a = 1 + gr_y$ and $b_j^{(k)} = gr_j^{(k)}$ for some $r_y, r_j^{(k)} \in R$.)
  - Sample $h \hookleftarrow D_{R,\sqrt{q}}$ and define the zero-testing parameter $p_{zt} = [\frac{h}{g} z^\kappa]_q \in R_q$.
  - Return public parameters $\mathrm{par} = (n, q, \sigma', m_r, y, \{x_j^{(k)}\}_{j \leq m_r, k \leq \kappa})$ and p_{zt}.
- **Level-0 sampler** $\mathsf{samp}(\mathrm{par})$: Sample $e \hookleftarrow D_{R,\sigma'}$ and return e.
  (Note that $e = e_L + ge_H$ for some unique coset representative $e_L \in \mathcal{P}_g$, and some $e_H \in R$.)
- **Level-k encoding** $\mathsf{enc}_k(\mathrm{par}, e)$: Given level-0 encoding $e \in R$ and parameters par:
  - Encode e at level k: Compute $u' = [e \cdot y^k]_q$.
  - Re-randomize: Sample $\rho_j \hookleftarrow \chi_k$ for $j \leq m_r$ and return $u = [u' + \sum_{j=1}^{m_r} \rho_j x_j^{(k)}]_q$.
    (Note that $u' = [c'/z^k]_q$ with $c' \in e_L + \mathcal{I}$ and $u = [(c' + \sum_j \rho_j b_j^{(k)})/z^k]_q$.)
- **Adding encodings** add: Given level-k encodings $u_1 = [c_1/z^k]_q$ and $u_2 = [c_2/z^k]_q$:
  - Return $u = [u_1 + u_2]_q$, a level-k encoding of $[c_1 + c_2]_g$.
- **Multiplying encodings** mult: Given level-k_1 encoding $u_1 = [c_1/z^{k_1}]_q$ and a level-k_2 encoding $u_2 = [c_2/z^{k_2}]_q$:
  - Return $u = [u_1 \cdot u_2]_q$, a level-$(k_1 + k_2)$ encoding of $[c_1 \cdot c_2]_g$.
- **Zero testing at level** κ $\mathsf{isZero}(\mathrm{par}, p_{zt}, u)$: Given a level-$\kappa$ encoding $u = [c/z^\kappa]_q$, return 1 if $\|[p_{zt} u]_q\|_\infty < q^{3/4}$ and 0 else.
  (Note that $[p_{zt} \cdot u]_q = [hc/g]_q$.)
- **Extraction at level** κ $\mathsf{ext}(\mathrm{par}, p_{zt}, u)$: Given a level-$\kappa$ encoding $u = [c/z^\kappa]_q$, return $v = \mathrm{MSB}_\ell([p_{zt} \cdot u]_q)$.
  (Note that if $c = [c]_g + gr$ for some $r \in R$, then $v = MSB_\ell(\frac{h}{g}([c]_g + gr)) = MSB_\ell(\frac{h}{g}[c]_g + hr)$, which is equal to $MSB_\ell(\frac{h}{g}[c]_g)$, with probability $1 - \lambda^{-\omega(1)}$.)

Fig. 1. The GGH graded encoding scheme.

The aim of isZero is to test whether the input $u = [c/z^\kappa]_q$ is a level-κ encoding of 0 or not, i.e., whether $c = g \cdot r$ for some $r \in R$. The following conditions ensure correctness of isZero, when $\chi_k = D_{\mathbb{Z},\sigma_k^*}$ (for all $k \leq \kappa$): the first one implies that false negatives do not exist (if u is

level-κ encoding of 0, then $\mathsf{isZero}(u)$ returns 1), whereas the second one implies that false positives occur with negligible probability (see Appendix A).

$$q > \max((n\ell_{g^{-1}})^8, ((m_r+1)\cdot n^{1.5}\sigma_1^*\sigma')^{8\kappa}) \quad (1)$$

$$q > (2n\sigma)^4. \quad (2)$$

The aim of ext is to extract a quantity from its input $u = [c/z^\kappa]_q$ that depends only on the encoded value $[c]_g$, but not on the randomizers. To avoid trivial solutions, one requires that this extracted value has min-entropy $\geq 2\lambda$ (if that is the case, then one can obtain a uniform distribution on $\{0,1\}^\lambda$, using a strong randomness extractor). The following two inequalities guarantee these properties, when $\chi_k = D_{\mathbb{Z},\sigma_k^*}$ (for all k). The first one implies that $\varepsilon_{ext} = \Pr[\mathsf{ext}(u) \neq \mathsf{ext}(u')]$ is negligible, when u and u' encode the same value $[c]_g$, whereas the second one provides large min-entropy (see Appendix A).

$$\frac{1}{4}\log q - \log(\frac{2n}{\varepsilon_{ext}}) \;\geq\; \ell \;\geq\; \log(8n\sigma). \quad (3)$$

3.2 The GDDH, GCDH and Ext-GCDH problems

The computational problems that are required to be hard for the GGH scheme depend on the application. Here we recall the definitions of the Graded Decisional and Computational Diffie-Hellman (GDDH and GCDH) problems from [10]. We introduce another natural variant that we call the Extraction Graded Computational Diffie-Hellman (Ext-GCDH), in which the goal is to compute the extracted string of a Diffie-Hellman encoding.

Definition 3.1 (GCDH/Ext-GCDH/GDDH). *The problems GCDH, Ext-GCDH and GDDH are defined as follows with respect to experiment of Figure 2:*[3]

- ***κ-graded CDH problem (GCDH):*** *On inputs* par, p_{zt} *and the* u_i*'s of Step 2, output a level-κ encoding of* $\prod_{i\geq 0} e_i + \mathcal{I}$*, i.e.,* $w \in R_q$ *such that* $\|[p_{zt}(v_C - w)]_q\| < q^{3/4}$.
- ***Extraction κ-graded CDH problem (Ext-GCDH):*** *On inputs* par, p_{zt} *and the* u_i*'s of Step 2, output the extracted string for a level-κ encoding of* $\prod_{i\geq 0} e_i + \mathcal{I}$*, i.e., the string* $w = \mathsf{ext}(\mathrm{par}, p_{zt}, v_C) = MSB_\ell([p_{zt}\cdot v_C]_q)$.
- ***κ-graded DDH problem (GDDH):*** *Distinguish between* v_D *and* v_R*, i.e., between the distributions* $\mathcal{D}_{DDH} = \{\mathrm{par}, p_{zt}, (u_i)_{0\leq i\leq\kappa}, v_D\}$ *and* $\mathcal{D}_R = \{\mathrm{par}, p_{zt}, (u_i)_{0\leq i\leq\kappa}, v_R\}$.

Ext-GCDH is at least as hard as GDDH: given v_x with $x \in \{\mathrm{DDH}, \mathrm{R}\}$, use the Ext-GCDH oracle to compute $w = \mathsf{ext}(\mathrm{par}, p_{zt}, v_C)$. Nevertheless, we show that it suffices for instantiating, in the random oracle model, at least some of the interesting applications of graded encoding schemes, at a higher efficiency than the instantiations of [10] based on GDDH.

3.3 The GGH re-randomization security requirement

The encoding re-randomization step in the GGH scheme is necessary for the hardness of the problems above. In [10], Garg et al. imposed the informal requirement that the re-randomization process "erases" the structure of the input encoding, while preserving the encoded coset. In setting parameters, they interpreted this requirement in the following natural way.

[3] Note that we use a slightly different process from [10], by adding a re-randomization to the element v_D. Without it, there exists a "division attack" against GDDH.

Given parameters λ, κ, proceed as follows:

1. Run $\mathsf{InstGen}(1^\lambda, 1^\kappa)$ to get $\text{par} = (n, q, \sigma', m_r, y, \{x_j^{(k)}\}_{j,k})$ and p_{zt}.
2. For $i = 0, \ldots, \kappa$:
   - Sample $e_i \hookleftarrow D_{R,\sigma'}$, $f_i \hookleftarrow D_{R,\sigma'}$,
   - Set $u_i = [e_i \cdot y + \sum_j \rho_{ij} x_j]_q$ with $\rho_{ij} \hookleftarrow \chi_1$ for all j.
3. Set $u^* = \left[\prod_{i=1}^{\kappa} u_i\right]_q$.
4. Set $v_C = [e_0 u^*]_q$.
5. Sample $\rho_j \hookleftarrow \chi_\kappa$ for all j, set $v_D = [e_0 u^* + \sum_j \rho_j x_j^{(\kappa)}]_q$.
6. Set $v_R = [f_0 u^* + \sum_j \rho_j x_j^{(\kappa)}]_q$.

Fig. 2. The GGH security experiment.

Given parameters $\lambda, \kappa, (\sigma_k^*)_{k \le \kappa}$, proceed as follows:

1. Run $\mathsf{InstGen}(1^\lambda, 1^\kappa)$ to get $\text{par} = (n, q, \sigma', m_r, y, \{x_j^{(k)}\}_{j,k})$ and p_{zt}. Write $x_j^{(k)} = [b_j^{(k)} z^{-k}]_q$ and $B^{(k)} = [b_1^{(k)}, \cdots, b_{m_r}^{(k)}] \in \mathcal{I}^{m_r}$.
2. For $i = 0, \ldots, \kappa$:
   - Sample $e_i \hookleftarrow U(R_g)$, $f_i \hookleftarrow U(R_g)$,
   - Set $u_i = [c_i z^{-1}]_q \hookleftarrow D_{can}^{(1)}(e_i)$, i.e., with $c_i \hookleftarrow D_{\mathcal{I}+e_i, \sigma_1^*(B^{(1)})^T}$.
3. Set $u^* = \left[\prod_{i=1}^{\kappa} u_i\right]_q$.
4. Set $v_C = [e_0 u^*]_q$.
5. Set $v_D = [c_D \cdot z^{-\kappa}]_q \hookleftarrow D_{can}^{(\kappa)}(\prod_{i=0}^{\kappa} e_i)$, i.e., with $c_D \hookleftarrow D_{\mathcal{I}+\prod_{i=0}^{\kappa} e_i, \sigma_\kappa^*(B^{(\kappa)})^T}$.
6. Set $v_R = [c_R \cdot z^{-\kappa}]_q \hookleftarrow D_{can}^{(\kappa)}(f_0 \prod_{i=1}^{\kappa} e_i)$, i.e., with $c_R \hookleftarrow D_{\mathcal{I}+f_0 \prod_{i=1}^{\kappa} e_i, \sigma_\kappa^*(B^{(\kappa)})^T}$.

Fig. 3. The canonical security experiment.

Definition 3.2 (Strong re-randomization security requirement). *Let $u' = [c'/z^k]_q$, with $c' = e_L + gr'$ be a fixed level-k encoding of $e_L \in R_g$, and let $u = [u' + \sum_j \rho_j x_k^{(j)}]_q = [c/z^k]_q$ with $c = e_L + gr$ and $r = r' + \sum_j \rho_j r_j^{(k)}$ be the re-randomized encoding, with $\rho_j \hookleftarrow \chi_k$ for $j \le m_r$. Let $D_u^{(k)}(e_L, r')$ denote the distribution of u (over the randomness of ρ_j's), parameterized by (e_L, r') and let $D_{\text{can}}^{(k)}(e_L)$ denote some canonical distribution, parameterized by e_L, that is independent of r'. Then we say that the* strong *re-randomization security requirement is satisfied at level k with respect to $D_{\text{can}}^{(k)}(e_L)$ and encoding norm $\gamma^{(k)}$ if $\Delta(D_u^{(k)}(e_L, r'), D_{\text{can}}^{(k)}(e_L)) \le 2^{-\lambda}$ for any $u' = [c'/z^k]_q$ with $\|c'\| \le \gamma^{(k)}$.*

The authors of [10] argued that with $\chi_k = D_{\mathbb{Z}, \sigma_k^*}$ (for $k \le \kappa$) and a "drowning ratio" $\sigma_k^*/\|r'\|$ exponential in security parameter λ, the distribution $D_u^{(k)}(e_L, r')$ is within negligible statistical distance to the canonical distribution $D_{\text{can}}^{(k)}(e_L) = [D_{\mathcal{I}+e_L, \sigma_k^*(B^{(k)})^T} \cdot z^{-k}]_q$. This requirement may be stronger than needed. Accordingly, we now clarify the desired goal.

3.4 Our security goal: canonical assumptions

We formalize a re-randomization security goal to capture a security guarantee against "statistical correlation" attacks on GCDH/Ext-GCDH/GDDH. We define *canonical variants* cGCDH/Ext-cGCDH/cGDDH of GCDH/Ext-GCDH/GDDH, using Figure 3. The main difference with Figure 2 is that the encodings $u_i = [c_i/z]_q$ of the hidden elements e_i, are sampled from a canonical distribution $D_{\text{can}}^{(1)}(e_i)$, parameterized by e_i, whose statistical parameters are independent of the encoded coset e_i, so that it is "by construction" immune against statistical correlation attacks. In particular, in the canonical distribution $D_{\text{can}}^{(1)}(e_i)$ that we use, c_i is sampled from a discrete Gaussian distribution $D_{\mathcal{I}+e_i, \sigma_1^*(B^{(1)})^T}$ (over the choice of the randomization, for a fixed e_i), whose statistical parameters such as center (namely 0) and deviation matrix $\sigma_1^*(B^{(1)})^T$ are independent of e_i. The only dependence this distribution has on the encoded element e_i is via its support $\mathcal{I} + e_i$.

We believe the canonical problems are cleaner and more natural than the non-canonical variants, since they decouple the re-randomization aspect from the rest of the computational problem. As a further simplification, the canonical variants also have their level-0 elements e_i distributed uniformly on R_g (rather than as reductions mod $\mathcal{I}$ of Gaussian samples).

Definition 3.3 (cGCDH/Ext-cGCDH/cGDDH). *The canonical problems cGCDH, Ext-cGCDH and cGDDH are defined as follows with respect to the experiment of Figure 3 and canonical encoding distribution $D^{(k)}_{can}(e)$ (parameterized by encoding level k and encoded element e):*

- ***cGCDH:*** *On inputs* par*, p_{zt} and the u_i's, output $w \in R_q$ such that $\|[p_{zt}(v_C - w)]_q\| < q^{3/4}$.*
- ***Ext-cGCDH:*** *On inputs* par*, p_{zt} and the u_i's, output: $w = \mathsf{ext}(\text{par}, p_{zt}, v_C) = MSB_\ell([p_{zt} \cdot v_C]_q)$.*
- ***cGDDH:*** *Distinguish between $\mathcal{D}_{DDH} = \{\text{par}, p_{zt}, (u_i)_{0 \le i \le \kappa}, v_D\}$ and $\mathcal{D}_R = \{\text{par}, p_{zt}, (u_i)_{0 \le i \le \kappa}, v_R\}$.*

Remark. One could consider alternative definitions of natural canonical encoding distributions besides the ones we adopt here. For instance, our results in this paper can also be adapted to hold for the canonical distribution $D^{(1)}_{\text{can}}(e_i)$ of $u_i = [c_i/z]_q$ in which c_i is sampled from $D_{\mathcal{I}+e_i,\sigma_1^*(B^{(1)})^T,e_i}$. In this alternative, although the center of c_i's distribution depends on e_i, the distribution of the randomizer r in the representation $c_i = e_i + g \cdot r$, is independent of e_i.

Given the canonical problems on whose hardness we wish to rely, our security goal for re-randomization with respect to the GCDH (resp. Ext-GCDH/GDDH) problems can now be easily formulated: hardness of the latter should be implied by hardness of the former.

Definition 3.4 (Re-randomization security goal). *We say that the re-randomization security goal is satisfied with respect to GCDH (resp. Ext-GCDH/GDDH) if any adversary against GCDH (resp. Ext-GCDH/ GDDH) with run-time $T = O(2^\lambda)$ and advantage $\varepsilon = \Omega(2^{-\lambda})$ can be used to construct an adversary against cGCDH (resp. Ext-cGCDH/cGDDH) with run-time $T' = \text{poly}(T, \lambda)$ and advantage $\varepsilon' = \Omega(\text{poly}(\varepsilon, \lambda))$.*

To set the background for our result, we show (in appendix) that Definition 3.2 implies that our security goal is reached. This is implicit in [10]. Looking ahead, we will show that in some cases, we may circumvent the strong re-randomization requirement of Definition 3.2 by replacing it with a weaker requirement (see Definition 6.1), while still reaching the security goal if Definition 3.4, with substantial consequent efficiency gains.

4 Polynomial drowning via Rényi divergence

In this section, we present our first result towards our improvement of the GGH scheme re-randomization. It shows that one may reduce the re-randomization "drowning" ratio $\sigma_k^*/\|r'\|$ from exponential to polynomial in the security parameter λ. Although the SD between the re-randomized encoding distribution D_1 (essentially a discrete Gaussian with an added offset vector r') and the desired canonical encoding distribution D_2 (a discrete Gaussian without an added offset vector) is then non-negligible, we show that these encoding distributions are still sufficiently close with respect to an alternative closeness measure to the SD, in the sense that switching between them preserves the success probability of any search problem adversary receiving these encodings as input, up to a polynomial transformation. This allows us to show that our re-randomization goal is satisfied for the search problems GCDH and Ext-GCDH.

Technically, the closeness measure we study is the *Rényi divergence* $R(D_1\|D_2)$ between the distributions D_1 and D_2, defined as the expected value of $D_1(r)/D_2(r)$ over the randomness of r sampled from D_1 (for brevity we will call $R(D_1\|D_2)$ the RD between D_1 and D_2). Intuitively, the RD is an alternative to SD as measure of distribution closeness, where we replace the *difference* between the distributions in SD, by the *ratio* of the distributions in RD. Accordingly, one may hope RD to have analogous properties to SD, where addition in the property of SD is replaced by multiplication in the analogous property of RD. Remarkably, this holds true in some sense, and we explore some of this below. In particular, a very important property of the SD is *probability preservation*: for any two distributions D_1, D_2 on space X, and any event $E \subseteq X$, we have $D_2(E) \geq D_1(E) - \Delta(D_1, D_2)$. Lyubashevsky et al. [18] observed an analogous property of the RD that follows roughly the above intuition: $D_2(E) \geq D_1(E)^2/R(D_1\|D_2)$. The latter property implies that as long as $R(D_1\|D_2)$ is bounded as $\text{poly}(\lambda)$, any event E of non-negligible probability $D_1(E)$ under D_1 will also have non-negligible probability $D_2(E)$ under D_2. We show that for our discrete Gaussian distributions D_1, D_2 above, we have $R(D_1\|D_2) = O(\text{poly}(\lambda))$, if $\sigma_k^*/\|r'\| = \Omega(\text{poly}(\lambda))$, as required for our re-randomization security goal.

4.1 The Rényi divergence (RD) and its properties

We review the RD [24,9] and some of its properties. For convenience, our definition of the RD is the exponential of the usual definition used in information theory [9], and coincides with a discrete version of the quantity R defined for continuous density functions in [18, Claim 5.11].

For any two discrete probability distributions P and Q such that $\text{Supp}(P) \subseteq \text{Supp}(Q)$ over a domain X and $\alpha > 1$, we define the Rényi divergence of orders α and ∞ by

$$R_\alpha(P\|Q) = \left(\sum_{x\in X} \frac{P(x)^\alpha}{Q(x)^{\alpha-1}}\right)^{\frac{1}{\alpha-1}} \quad \text{and} \quad R_\infty(P\|Q) = \max_{x\in X} \frac{P(x)}{Q(x)},$$

with the convention that the fraction is zero when both numerator and denominator are zero. A convenient choice for computations (as also used in [18]) is $\alpha = 2$, in which case we omit the α subscript. Note that $R_\alpha(P\|Q)^{\alpha-1} = \sum_x P(x) \cdot (P(x)/Q(x))^{\alpha-1} \leq R_\infty(P\|Q)^{\alpha-1}$. We list several properties of the RD that can be considered the multiplicative analogues of those of the SD.

Lemma 4.1. *Let P_1, P_2, P_3 and Q_1, Q_2 denote discrete distributions on a domain X and let $\alpha \in (1, \infty]$. Then the following properties hold:*

- ***Log. Positivity:*** *$R_\alpha(P_1\|Q_1) \geq R_\alpha(P_1\|P_1) = 1$.*
- ***Data Processing Inequality:*** *$R_\alpha(P_1^f\|Q_1^f) \leq R_\alpha(P_1\|Q_1)$ for any function f, where P_1^f (resp. Q_1^f) denotes the distribution of $f(y)$ induced by sampling $y \hookleftarrow P_1$ (resp. $y \hookleftarrow Q_1$).*
- ***Multiplicativity:*** *Let P and Q denote any two distributions of a pair of random variables (Y_1, Y_2) on $X \times X$. For $i \in \{1, 2\}$, assume P_i (resp. Q_i) is the marginal distribution of Y_i under P (resp. Q), and let $P_{2|1}(\cdot|y_1)$ (resp. $Q_{2|1}(\cdot|y_1)$) denote the conditional distribution of Y_2 given that $Y_1 = y_1$. Then we have:*
  - *$R_\alpha(P\|Q) = R_\alpha(P_1\|Q_1) \cdot R_\alpha(P_2\|Q_2)$ if Y_1 and Y_2 are independent.*
  - *$R_\alpha(P\|Q) \leq R_\infty(P_1\|Q_1) \cdot \max_{y_1\in X} R_\alpha(P_{2|1}(\cdot|y_1)\|Q_{2|1}(\cdot|y_1))$.*
- ***Weak Triangle Inequality:*** *We have:*

$$R_\alpha(P_1\|P_3) \leq \begin{cases} R_\alpha(P_1\|P_2) \cdot R_\infty(P_2\|P_3), \\ R_\infty(P_1\|P_2)^{\frac{\alpha}{\alpha-1}} \cdot R_\alpha(P_2\|P_3). \end{cases}$$

– R_∞ ***Triangle Inequality:*** *If $R_\infty(P_1\|P_2)$ and $R_\infty(P_2\|P_3)$ are defined, then $R_\infty(P_1\|P_3) \leq R_\infty(P_1\|P_2) \cdot R_\infty(P_2\|P_3)$.*
– ***Probability Preservation:*** *Let $A \subseteq X$ be an arbitrary event. Then $Q_1(A) \geq P_1(A)^{\frac{\alpha}{\alpha-1}} / R_\alpha(P_1\|Q_1)$.*

Proof. The log. positivity and data processing inequalities are proved in [9, Th. 8&9].

For multiplicativity, we have

$$R_\alpha(P\|Q)^{\alpha-1} = \sum_{x_1,x_2} \frac{(P_1(x_1) \cdot P_{2|1}(x_2|x_1))^\alpha}{(Q_1(x_1) \cdot Q_{2|1}(x_2|x_1))^{\alpha-1}} = \sum_{x_1} \frac{P_1(x_1)^\alpha}{Q_1(x)^{\alpha-1}} \cdot R_\alpha(P_{2|1}(\cdot|x_1)\|Q_{2|1}(\cdot|x_1))^{\alpha-1}.$$

If X_1 and X_2 are independent, then we have $P_{2|1}(x_2|x_1) = P_2(x_2)$ and $Q_{2|1}(x_2|x_1) = Q_2(x_2)$ for all x_1, and the result follows. More generally, since $R_\alpha(P\|Q)^{\alpha-1}$ is the expected value of $f(x_1) = \frac{P_1(x_1)^{\alpha-1}}{Q_1(x)^{\alpha-1}} \cdot R_\alpha(P_{2|1}(\cdot|x_1)\|Q_{2|1}(\cdot|x_1))^{\alpha-1}$ over x_1 sampled from P_1, it follows that $R_\alpha(P\|Q)^{\alpha-1} \leq \max_{x_1} f(x_1)$, which gives the second multiplicativity property.

For the first weak triangle inequality, we have

$$R_\alpha(P_1\|P_3)^{\alpha-1} = \sum_x \frac{P_1(x)^\alpha}{P_3(x)^{\alpha-1}} = \sum_x \frac{P_1(x)^\alpha}{P_2(x)^{\alpha-1}} \cdot \frac{P_2(x)^{\alpha-1}}{P_3(x)^{\alpha-1}} \leq \left(\sum_x \frac{P_1(x)^\alpha}{P_2(x)^{\alpha-1}}\right) \cdot \max_x \frac{P_2(x)^{\alpha-1}}{P_3(x)^{\alpha-1}},$$

which gives the desired result. Similarly, for the second weak triangle inequality,

$$R_\alpha(P_1\|P_3)^{\alpha-1} = \sum_x \frac{P_1(x)^\alpha}{P_3(x)^{\alpha-1}} = \sum_x \frac{P_1(x)^\alpha}{P_2(x)^\alpha} \cdot \frac{P_2(x)^\alpha}{P_3(x)^{\alpha-1}} \leq \left(\max_x \frac{P_1(x)^\alpha}{P_2(x)^\alpha}\right) \cdot \sum_x \frac{P_2(x)^\alpha}{P_3(x)^{\alpha-1}},$$

as required. For the R_∞ triangle inequality, we have

$$R_\infty(P_1\|P_3) = \max_x \frac{P_1(x)}{P_3(x)} = \max_x \frac{P_1(x)}{P_2(x)} \cdot \frac{P_2(x)}{P_3(x)} \leq \left(\max_x \frac{P_1(x)}{P_2(x)}\right) \cdot \left(\max_x \frac{P_2(x)}{P_3(x)}\right).$$

Finally, the probability preservation property is proved in [18, Claim 5.11] for the case $\alpha = 2$ using the Cauchy-Schwarz inequality. The general case follows by replacing the latter with the more general Holder inequality, which states that $\sum_{x\in A} |f(x)g(x)| \leq (\sum_{x\in A} |f(x)|^p)^{1/p} \cdot (\sum_{x\in A} |g(x)|^{1/(1-1/p)})^{1-1/p}$ for real-valued functions f, g and $p \geq 1$. Taking $f(x) = \frac{P_1(x)}{Q_1(x)^{1-1/\alpha}}$, $g(x) = Q_1(x)^{1-1/\alpha}$, and $p = \alpha$, we get $P_1(A) \leq (\sum_{x\in A} \frac{P_1(x)^\alpha}{Q_1(x)^{\alpha-1}})^{1/\alpha} \cdot Q_1(A)^{1-1/\alpha}$. The fact that $\sum_{x\in A} \frac{P_1(x)^\alpha}{Q_1(x)^{\alpha-1}} \leq R_\alpha(P_1\|Q_1)^{\alpha-1}$ provides the result. □

We note that the RD does not satisfy the (multiplicative) triangle inequality $R(P_1\|P_3) \leq R(P_1\|P_2) \cdot R(P_2\|P_3)$ in general (see [9]), but a weaker inequality holds if one of the pairs of distributions has a bounded R_∞ divergence, as shown above. We also observe that R_∞ *does* satisfy the triangle inequality.

4.2 The Rényi divergence between a discrete Gaussian and its offset

For our re-randomization application, we are interested in the RD between two discrete Gaussians with the same deviation matrix S, that differ by some fixed offset vector d. The following result shows that their RD is $O(1)$ if $\sigma_n(S)/\|d\| = \Omega(1)$.

Lemma 4.2. *For any n-dimensional lattice $\Lambda \subseteq \mathbb{R}^n$ and rank n matrix $S \in \mathbb{R}^{m\times n}$ (with$4\geq n$), let P be the distribution $D_{\Lambda,S,w}$ and Q be the distribution $D_{\Lambda,S,z}$ for some fixed $w, z \in \mathbb{R}^n$. If $w, z \in \Lambda$, let $\varepsilon = 0$. Otherwise, fix $\varepsilon \in (0,1)$ and assume that $\sigma_n(S) \geq \eta_\varepsilon(\Lambda)$. Then:*

$$\begin{aligned} R(P\|Q) &\in \left[\left(\frac{1-\varepsilon}{1+\varepsilon}\right)^2, \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^2\right] \cdot \exp(2\pi\|S^{-T}(w-z)\|^2) \\ &\subseteq \left[\left(\frac{1-\varepsilon}{1+\varepsilon}\right)^2, \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^2\right] \cdot \exp\left(\frac{2\pi\|w-z\|^2}{\sigma_n(S)^2}\right). \end{aligned}$$

Proof. By definition,

$$P(x) = \frac{\exp(-\pi\|(S^T)^\dagger(x-w)\|^2)}{\sum_{x\in\Lambda}\exp(-\pi\|(S^T)^\dagger(x-w)\|^2)} \quad \text{and} \quad Q(x) = \frac{\exp(-\pi\|(S^T)^\dagger(x-z)\|^2)}{\sum_{x\in\Lambda}\exp(-\pi\|(S^T)^\dagger(x-z)\|^2)}.$$

We have:

$$\begin{aligned} R(P\|Q) &= \sum_{x\in\Lambda}\frac{P(x)^2}{Q(x)} \\ &= \frac{\sum_{y\in\Lambda}\exp(-\pi\|(S^T)^\dagger(y-z)\|^2)}{(\sum_{y\in\Lambda}\exp(-\pi\|(S^T)^\dagger(y-w)\|^2))^2} \cdot \sum_{x\in\Lambda}\exp(-2\pi\|(S^T)^\dagger(x-w)\|^2 + \pi\|(S^T)^\dagger(x-z)\|^2). \end{aligned}$$

Defining $c = 2w - z$, we have that:

$$2\|(S^T)^\dagger(x-w)\|^2 - \|(S^T)^\dagger(x-z)\|^2 = \|(S^T)^\dagger(x-c)\|^2 - 2\|(S^T)^\dagger(w-z)\|^2.$$

Hence,

$$R(P\|Q) = \exp(2\pi\|(S^T)^\dagger(w-z)\|^2) \cdot \frac{\sum_{x\in\Lambda}\exp(-\pi\|(S^T)^\dagger(x-c)\|^2) \cdot \sum_{y\in\Lambda}\exp(-\pi\|(S^T)^\dagger(y-z)\|^2)}{(\sum_{y\in\Lambda}\exp(-\pi\|(S^T)^\dagger(y-w)\|^2))^2}.$$

Notice that for any $z \in \Lambda$, we have $\sum_{x\in\Lambda}\exp(-\pi\|(S^T)^\dagger(x-z)\|^2) = \sum_{x\in\Lambda}\exp(-\pi\|(S^T)^\dagger x\|^2)$. From this, we conclude that if $w, z \in \Lambda$, then $c \in \Lambda$ and hence the sums in the quotient above cancel out, and we get $R(P\|Q) = \exp(2\pi\|(S^T)^\dagger(w-z)\|^2)$. In general, for any $y, z \in \mathbb{R}^n$, we have

$$\sum_{y\in\Lambda}\exp(-\pi\sigma_1((S^T)^\dagger)^2 \cdot \|y-z\|^2) \leq \sum_{y\in\Lambda}\exp(-\pi\|(S^T)^\dagger \cdot (y-z)\|^2) \leq \sum_{y\in\Lambda}\exp(-\pi\sigma_n((S^T)^\dagger)^2 \cdot \|y-z\|^2),$$

using the fact that $\sigma_n((S^T)^\dagger) \cdot \|y-z\| \leq \|(S^T)^\dagger \cdot (y-z)\| \leq \sigma_1((S^T)^\dagger) \cdot \|y-z\|$. But

$$\begin{aligned} \sum_{y\in\Lambda}\exp(-\pi\sigma_1((S^T)^\dagger)^2 \cdot \|y-z\|^2) &= \rho_{1/\sigma_1((S^T)^\dagger),z}(\Lambda) = \rho_{\sigma_n(S),z}(\Lambda) \\ \sum_{y\in\Lambda}\exp(-\pi\sigma_n((S^T)^\dagger)^2 \cdot \|y-z\|^2) &= \rho_{1/\sigma_n((S^T)^\dagger),z}(\Lambda) = \rho_{\sigma_1(S),z}(\Lambda). \end{aligned}$$

Using the assumption $\sigma_1(S) \geq \sigma_n(S) \geq \eta_\varepsilon(\Lambda)$ and Lemma 2.5, it follows that $\rho_{\sigma_1(S),z}(\Lambda)$ and $\rho_{\sigma_n(S),z}(\Lambda)$ are both in the interval $[1-\varepsilon, 1+\varepsilon] \cdot (\det\Lambda)^{-1}$. From the above inequality, we get that $\sum_{y\in\Lambda}\exp(-\pi\|(S^T)^\dagger \cdot (y-z)\|^2)$ is also in this interval. Applying this to the sums in the expression for $R(P\|Q)$ gives the claimed interval for $R(P\|Q)$.

The claimed inequality follows from $\|(S^T)^\dagger z\|^2 \leq \sigma_1((S^T)^\dagger)^2 \cdot \|z\|^2$ and $\sigma_1((S^T)^\dagger) = 1/\sigma_n(S)$. □

5 A discrete Gaussian leftover hash lemma over R

In this section, we present our second main result for improving the GGH scheme re-randomization algorithm. Recall that the GGH algorithm re-randomizes a level-k encoding u' into $u = [u' + \sum_{j=1}^{m_r} \rho_j x_j^{(k)}]_q$, where the ρ_j's are sampled from $\chi_k = D_{\mathbb{Z},\sigma_k^*}$ and $x_j^{(k)} = [b_j^{(k)}/z^k]_q = [g r_j^{(k)}/z^k]_q$. To show that the distribution of $\sum_{j=1}^{m_r} \rho_j b_j^{(k)}$ is close to a discrete Gaussian over $\mathcal{I}$, they then apply the discrete Gaussian LHL from [2, Th. 3], using $m_r = \Omega(n \log n)$ fixed elements $b_j^{(k)} \in \mathcal{I}$ that are published obliviously as randomizers "inside" the public zero-encodings $x_j^{(k)}$. We show that it suffices to sample 2 randomizers as elements of the full n-dimensional ring R, rather than just from $\mathbb{Z}$, i.e., we set $\chi_k = D_{R,\sigma_k^*}$. In Appendix B, we review the results of [2], as our proof follows the same high-level steps.

For a fixed $X = (x_1, x_2) \in R^2$, we define the distribution $\widetilde{\mathcal{E}}_{X,s} = x_1 D_{R,s} + x_2 D_{R,s}$ as the distribution induced by sampling $\boldsymbol{u} = (u_1, u_2) \in R^2$ from a discrete spherical Gaussian with parameter s, and outputting $y = x_1 u_1 + x_2 u_2$. We prove the following result on $\widetilde{\mathcal{E}}_{X,s}$.

Theorem 5.1. *Let $R = \mathbb{Z}[x]/\langle x^n + 1\rangle$ with n a power of 2 and $\mathcal{I} = \langle g \rangle \subseteq R$, for some $g \in R$. Fix $\varepsilon \in (0, 1/2)$, $X = (x_1, x_2) \in \mathcal{I} \times \mathcal{I}$ and $s > 0$ satisfying the conditions*

- ***Column span:*** *$X \cdot R^2 = \mathcal{I}$.*
- ***Smoothing:*** *$s \geq \max(\|g^{-1} x_1\|_\infty, \|g^{-1} x_2\|_\infty) \cdot n \cdot \sqrt{2\log(2n(1 + 1/\varepsilon))/\pi}$.*

Then, for all $x \in \mathcal{I}$ we have $\widetilde{\mathcal{E}}_{X,s}(x) = c f(x) \cdot D_{\mathcal{I},sX^T}(x)$, for some constant c and function f with values in $[\frac{1-\varepsilon}{1+\varepsilon}, 1]$. In particular, we have

$$\Delta(\widetilde{\mathcal{E}}_{X,s}, D_{\mathcal{I},sX^T}) \leq 2\varepsilon \quad \textit{and} \quad \max(R_\infty(\widetilde{\mathcal{E}}_{X,s} \| D_{\mathcal{I},sX^T}), R_\infty(D_{\mathcal{I},sX^T} \| \widetilde{\mathcal{E}}_{X,s})) \leq 1 + 4\varepsilon.$$

Finally, if $s' \cdot \sigma_n(g^{-1}) \geq 7n^{1.5} \ln^{1.5}(n)$,[4] $x_1, x_2 \hookleftarrow D_{\mathcal{I},s'}$ and n grows to infinity, then the first condition holds with probability $\Omega(1)$.

We prove this result for $g = 1$, and then we generalize to general g. First, we consider the column span condition.

Lemma 5.2 (Adapted from [29, Le. 4.2 and Le. 4.4]). *Let $S \in \mathbb{R}^{n \times n}$, and $\sigma_n(S) \geq 7n^{1.5} \ln^{1.5}(n)$. For n going to infinity, we have $\Pr_{x_1,x_2 \hookleftarrow D_{R,S}}[X \cdot R^2 = R] \geq \Omega(1)$.*

Let $A_X \subseteq \{(v_1, v_2) \in R^2 : x_1 v_1 + x_2 v_2 = 0\}$ be the 1-dimensional R-module of vectors orthogonal to X. We view A_X as an n-dimensional lattice in $\mathbb{Z}^{2n}$, via the polynomial-to-coefficient-vector mapping.

Lemma 5.3 (Adapted from [2, Le. 10]). *Fix X such that $X \cdot R^2 = R$ and A_X as above. If $s \geq \eta_\varepsilon(A_X)$, then $\widetilde{\mathcal{E}}_{X,s}(z) = c f(z) \cdot D_{\mathbb{Z}^n,sX^T}(z)$ for any $z \in R$, for some constant c and function f with values in $[\frac{1-\varepsilon}{1+\varepsilon}, 1]$.[5] In particular, we have*

$$\Delta(\widetilde{\mathcal{E}}_{X,s}, D_{\mathbb{Z}^n,sX^T}) \leq \frac{\varepsilon}{1-\varepsilon} \quad \textit{and} \quad \max(R_\infty(\widetilde{\mathcal{E}}_{X,s} \| D_{\mathbb{Z}^n,sX^T}), R_\infty(D_{\mathbb{Z}^n,sX^T} \| \widetilde{\mathcal{E}}_{X,s})) \leq \frac{1+\varepsilon}{1-\varepsilon}.$$

[4] By abuse of notation, we identify $g^{-1} \in K$ with the linear map over $\mathbb{Q}^n$ obtained by applying the polynomial-to-coefficient-vector mapping to the map $r \mapsto g^{-1} r$.

[5] The normalization constant c was omitted in [2].

We now study the quantity $\eta_\varepsilon(A_X)$. First, we show that all successive Minkowski minima of A_X are equal. This property is inherited from the "equal minima property" of ideal lattices in R.

Lemma 5.4. *Let X and A_X be as above. Then $\lambda_1(A_X) = \cdots = \lambda_n(A_X)$.*

Proof. We observe that A_X is closed under scalar multiplication by an arbitrary element $w \in R$, i.e., if $\boldsymbol{v} = (v_1, v_2) \in A_X$ then $w \cdot \boldsymbol{v} = (w \cdot v_1, w \cdot v_2) \in A_X$. In particular, let $\boldsymbol{v} \in A_X$ be a vector of norm $\|\boldsymbol{v}\| = \lambda_1(A_X)$. For $i = 0, \ldots, n-1$, let $e_i(x) = x^i \in R$. Then the n vectors $(e_0 \cdot \boldsymbol{v}, \ldots, e_{n-1} \cdot \boldsymbol{v})$ are in A_X, and all have the same norm $\lambda_1(A_X)$, because $\|e_j \cdot v_i\| = \|v_i\|$ for all i, j. Further, these n vectors are linearly independent over $\mathbb{Q}$: let i be such that $v_i \neq 0$ (which must exist since $\boldsymbol{v} \neq \boldsymbol{0}$); the vectors $(e_0 \cdot v_i, \ldots, e_{n-1} \cdot v_i)$ are linearly independent over $\mathbb{Q}$, because the fraction field K of R is a field (it they were not linearly independent over $\mathbb{Q}$, we would have $(\sum_j \alpha_j e_j) \cdot v_i = 0$ for some non-zero $\alpha = \sum_j \alpha_j e_j \in K$). It follows that $\lambda_1(A_X) = \cdots = \lambda_n(A_X) = \|\boldsymbol{v}\|$. □

Lemma 5.5. *Let X and A_X be as above. Then we have $\eta_\varepsilon(A_X) \leq \max(\|x_1\|_\infty, \|x_2\|_\infty) \cdot n \cdot \sqrt{2\log(2n(1+1/\varepsilon))/\pi}$.*

Proof. We first use Lemma 5.4 and Minkowski's second theorem (see Lemma 2.1) on the lattice A_X:

$$\lambda_n(A_X) \;=\; \big(\textstyle\prod_{1 \leq i \leq n} \lambda_i(A_X)\big)^{1/n} \;\leq\; \sqrt{n} \cdot \big(\det(A_X)\big)^{1/n}.$$

Now, observe that $A_X = L_X^\perp$, where $L_X = R \cdot X = \{(r \cdot x_1, r \cdot x_2) : r \in R\}$ is viewed as a sublattice of $\mathbb{Z}^{2n}$. We have, by Lemma 2.2, that $\det(A_X) \leq \det(L_X) \leq \|X\|^n$, where the latter inequality follows from the Hadamard inequality, with $\|X\| = \sqrt{\|x_1\|^2 + \|x_2\|^2} \leq \max(\|x_1\|_\infty, \|x_2\|_\infty) \cdot \sqrt{2n}$. As a consequence $\lambda_n(A_X) \leq \max(\|x_1\|_\infty, \|x_2\|_\infty) \cdot \sqrt{2}n$. By Lemma 2.4, we have $\eta_\varepsilon(A_X) \leq \sqrt{\ln(2n(1+1/\varepsilon))/\pi} \cdot \lambda_n(A_X)$, which completes the proof. □

Combining the above lemmas, we get Theorem 5.1 for $g = 1$. The general case is proved as follows. The injective map $M_g : y \mapsto g \cdot y$ on R takes the distribution $\widetilde{\mathcal{E}}_{\overline{X},s}$ with $\overline{X} = g^{-1} \cdot X$ to the distribution $\widetilde{\mathcal{E}}_{X,s}$, while it takes $D_{R,s\overline{X}^T}$ to $D_{\mathcal{I},sX^T}$, with $\mathcal{I} = \langle g \rangle$. The conditions $X \cdot R^2 = \mathcal{I}$ and $\overline{X} \cdot R^2 = R$ are equivalent. The smoothing condition is satisfied for $\overline{X}$ by the choice of s. Thus we can apply Theorem 5.1 with $g = 1$ to $\widetilde{\mathcal{E}}_{\overline{X},s}$, and conclude by applying the mapping M_g to get the general case of Theorem 5.1. For the very last statement of Theorem 5.1, it suffices to observe that $D_{\mathcal{I},\beta} = g \cdot D_{R,s'(g^{-1})^T}$.[6] □

6 Our improved GGH grading scheme: GGHLite

We are now ready to describe our simpler and more efficient variant of the GGH grading scheme, that we call GGHLite. The scheme is summarized in Figure 4. The modifications from the original GGH scheme consist in:

- Using $m_r = 2$ re-randomization elements x_1, x_2 in the public key, sampling the randomizers ρ_1, ρ_2 from a discrete Gaussian D_{R,σ_1^*} over the whole ring R (rather than from $\mathbb{Z}$), applying our algebraic ring variant of the LHL from Section 5.
- Saving an exponential factor $\approx 2^\lambda$ in the re-randomization parameter σ_1^* by applying the RD bounds from Section 4.

[6] With the same abuse of notation as in the previous footnote, for the term $(g^{-1})^T$.

In terms of re-randomization security requirement, we relax the strong SD-based requirement on the original GGH scheme to the following weaker RD-based requirement on GGHLite.

Definition 6.1 (Weak re-randomization security requirement). *Using the notations of Definition 3.2, we say that the* weak *re-randomization security requirement is satisfied at level k with respect to $D^{(k)}_{\mathrm{can}}(e_L)$ and encoding norm $\gamma^{(k)}$ if $R(D^{(k)}_u(e_L, r')\|D^{(k)}_{\mathrm{can}}(e_L)) = O(\mathrm{poly}(\lambda))$ for any $u' = [c'/z^k]_q$ such that $\|c'\| \le \gamma^{(k)}$.*

We summarize GGHLite in Figure 4, which only shows the algorithms differing from those in the GGH scheme of Figure 1.

- **Instance generation** $\mathsf{InstGen}(1^\lambda, 1^\kappa)$: Given security parameter λ and multilinearity parameter κ, determine scheme parameters n, q, $m_r = 2$, σ, σ', $\ell_{g^{-1}}$, ℓ_b, ℓ, based on the scheme analysis. Then proceed as follows:
  - Sample $g \hookleftarrow D_{R,\sigma}$ until $\|g^{-1}\| \le \ell_{g^{-1}}$ and $\mathcal{I} = \langle g \rangle$ is a prime ideal and $\|g\| \le \sqrt{n} \cdot \sigma$.
  - Sample $z \hookleftarrow U(R_q)$.
  - Sample a level-1 encoding of 1: $y = [a \cdot z^{-1}]_q$ with $a \hookleftarrow D_{1+\mathcal{I},\sigma'}$.
  - For $k \le \kappa$:
    - Sample $B^{(k)} = (b^{(k)}_1, b^{(k)}_2)$ from $(D_{\mathcal{I},\sigma'})^2$. If $\langle b^{(k)}_1, b^{(k)}_2 \rangle \ne \mathcal{I}$, or $\sigma_n(\mathrm{rot}(B^{(k)})) < \ell_b$ or $\|B^{(k)}\| > \sqrt{n} \cdot \sigma'$, then re-sample.
    - Define level-k encodings of 0: $x^{(k)}_1 = [b^{(k)}_1 \cdot z^{-k}]_q$, $x^{(k)}_2 = [b^{(k)}_2 \cdot z^{-k}]_q$.
  - Sample $h \hookleftarrow D_{R,\sqrt{q}}$ and define the zero-testing parameter $p_{zt} = [\frac{h}{g} z^\kappa]_q \in R_q$.
  - Return public parameters $\mathrm{par} = (n, q, y, \{(x^{(k)}_1, x^{(k)}_2)\}_{k \le \kappa})$ and p_{zt}.
- **Level-k encoding** $\mathsf{enc}_k(\mathrm{par}, e)$: Given level-0 encoding $e \in R$ and parameters par:
  - Encode e at level k: Compute $u' = [e \cdot y^k]_q$.
  - Return $u = [u' + \rho_1 \cdot x^{(k)}_1 + \rho_2 \cdot x^{(k)}_2]_q$, with $\rho_1, \rho_2 \hookleftarrow D_{R,\sigma^*_k}$.

Fig. 4. The new algorithms of our GGHLite scheme.

Choice of σ, $\ell_{g^{-1}}$ and σ', ℓ_b. The upper bound $\ell_{g^{-1}}$ on $\|g^{-1}\|$ in the rejection test of InstGen can be chosen as small as possible while keeping the rejection probability p_g bounded from 1. According to Lemma 2.7 and Lemma 2.4 with $t = 2\sqrt{2\pi e} n p_g^{-1}$ and $\delta = 1/3$, one can choose

$$\ell_{g^{-1}} = 4\sqrt{\pi e n}/(p_g \sigma) \quad \text{and} \quad \sigma \ge 4\pi n \sqrt{\mathrm{e} \ln(8n)/\pi}/p_g, \tag{4}$$

to achieve $p_g < 1$. Note that the same choices apply to the GGH scheme: here we have a rigorous bound on p_g instead of the heuristic arguments for estimating in $\|g^{-1}\|$ in [10]; however, as in [10], we do not have a rigorous bound on the probability that $\mathcal{I}$ is prime conditioned on this choice.

Let p_b be the rejection probability for the lower bound ℓ_b on $\sigma_n(B^{(k)})$ in the rejection test of InstGen. To keep p_b away from 1, we use that $\sigma_n(B^{(k)})^2 = \min_{u \in K, \|u\|=1} \sum_{i=1,2} \|u \cdot b^{(k)}_i\|^2 \ge \sum_{i=1,2} \sigma_n(b^{(k)}_i)^2$. Applying Lemma 2.7 with $t = 2\sqrt{2\pi e} n p_b^{-1}$ and $\delta = 1/3$, we get that $\sigma_n(b^{(k)}_i) > \frac{p_b}{8\sqrt{\pi e n}} \cdot \sigma'$, except with probability $\le p_b$ for $i \in \{1, 2\}$ if $\sigma' \ge \frac{t}{\sqrt{2\pi}} \eta_{1/3}(\mathcal{I})$, where $\eta_{1/3}(\mathcal{I}) \le \sqrt{\ln(8n)/\pi} \cdot \|g\|$ by Lemma 2.4. Therefore, we can choose

$$\ell_b = \frac{p_b}{2\sqrt{\pi e n}} \cdot \sigma' \quad \text{and} \quad \sigma' \ge 2n^{1.5} \sigma \sqrt{\mathrm{e} \ln(8n)/\pi}/p_b. \tag{5}$$

We also need to bound the probability p'_b of the first rejection test $\langle b_1^{(k)}, b_2^{(k)} \rangle \neq \mathcal{I}$. This is bounded by some constant < 1 by Theorem 5.1, but it requires the assumption $\sigma' \cdot \sigma_n(g^{-1}) \geq 7n^{1.5}\ln^{1.5}(n)$. To use Theorem 5.1 to obtain a rigorous bound on p'_b, we can satisfy the assumption as follows. Using the lower bound $\sigma_n(g^{-1}) \geq \frac{1}{\sqrt{n}\|g\|}$ from the remark after Lemma 2.7, and using the rejection condition $\|g\| \leq \sqrt{n} \cdot \sigma$, we have $\sigma_n(g^{-1}) \geq \frac{1}{n\sigma}$, so the Theorem 5.1 assumption is satisfied by setting

$$\sigma' \geq 7n^{2.5}\ln^{1.5}(n) \cdot \sigma. \tag{6}$$

Zero-testing and extraction correctness. The correctness conditions for zero-testing and correctness remain the same as conditions (2), (3) for the original GGH scheme. The only modification needed is for condition (1), because in GGHLite, $m_r = 2$ and $\rho_j \in R$ so $\|\rho_j b_j^{(1)}\| \leq \sqrt{n}\|\rho_j\|\|b_j^{(1)}\|$. Accordingly, condition (1) is replaced by:

$$q > \max\left((n\ell_{g^{-1}})^8, (3 \cdot n^{1.5}\sigma^*\sigma')^{8\kappa}\right). \tag{7}$$

Security. We state our improved re-randomization security reduction for GGHLite, that works with much smaller parameters than GGH. To our knowledge, it is the first security proof in which the RD is used to replace the SD in a sequence of games, using the RD properties from Section 4 to combine the bounds on changes between games. This allows us to gain the benefits of RD over SD, for both the drowning and smoothing aspects. Namely, with $\varepsilon_d, \varepsilon_\rho, \varepsilon_e$ in Theorem 6.2 set as large as $O(\log\lambda/\kappa)$, our weak security requirement of Definition 6.1 is satisfied (the RD between real and canonical encoding distributions is bounded by the quantity $R = \text{poly}(\lambda)$ in Theorem 6.2), and our re-randomization goal for Ext-GCDH is achieved (whereas the strong requirement of Definition 3.2 is not satisfied).

Theorem 6.2 (Security of GGHLite). *Let $\varepsilon_d, \varepsilon_\rho, \varepsilon_e \in (0, 1/2)$ and $\kappa \leq 2^n$. Suppose that the following conditions are satisfied for* GGHLite*:*

- ***LHL Smoothing:***
$$\sigma_1^* \geq n^{1.5} \cdot \ell_{g^{-1}} \cdot \sigma' \cdot \sqrt{2\log(4n \cdot \varepsilon_\rho^{-1})/\pi}. \tag{8}$$
- ***Offset "Drowning:"***
$$\sigma_1^* \geq n^{1.5} \cdot (\sigma')^2 \cdot \sqrt{8\pi\varepsilon_d^{-1}}/\ell_b. \tag{9}$$
- samp ***Uniformity Smoothing:***
$$\sigma' \geq \sigma \cdot \sqrt{n\ln(4n \cdot \varepsilon_e^{-1})/\pi}. \tag{10}$$

Then, if $\mathcal{A}$ is an adversary against the (non-canonical) Ext-GCDH problem for GGHLite *with runtime T and advantage ε, then $\mathcal{A}$ is also an adversary against the canonical problem Ext-cGCDH for* GGHLite *with $T' = T$ and advantage*

$$\varepsilon' \geq (\varepsilon - O(\kappa \cdot 2^{-n}))^2/R \quad \textit{with} \quad R = 2^{O(\kappa\cdot(\varepsilon_d+\varepsilon_\rho+\varepsilon_e+2^{-n}))}. \tag{11}$$

In particular, there exist ε_d, ε_e, ε_ρ bounded as $O(\log\lambda/\kappa)$ such that the re-randomization security goal in Definition 3.4 is satisfied by GGHLite *with respect to problem Ext-GCDH.*

Proof. We consider a sequence of games $\mathsf{Game}_0, \ldots, \mathsf{Game}_5$, where the distributions of the view of $\mathcal{A}$ differ among the games as follows:

- Game_0: The Ext-GCDH experiment, where $y = [az^{-1}]_q$ with $a = 1 + gr_y \hookleftarrow D_{1+\mathcal{I},\sigma'}$ and $\mathcal{I} = \langle g \rangle$, $u_i = [(e_{i,L} + \sum_j \rho_{ij} b_j^{(1)} + c_i) \cdot z^{-1}]_q$ for $i \in \{0, \ldots, \kappa\}$, $e_{i,L} = [e_i]_g$, $e_i = e_{i,L} + ge_{i,H} \hookleftarrow D_{R,\sigma'}$, and $c_i = g(e_{i,L} r_y + e_{i,H}) + g^2 r_y e_{i,H}$.
- Game_1: Modification of Game_0 in which e_i (for $i \in \{0, \ldots, \kappa\}$) and a are sampled from the truncated tail Gaussians $D^t_{R,\sigma'}$ and $D^t_{1+\mathcal{I},\sigma'}$ (instead of the untruncated Gaussians $D_{R,\sigma'}$ and $D_{1+\mathcal{I},\sigma'}$ respectively).
- Game_2: Modification of Game_1 in which the distribution of the re-randomization term $\sum_j \rho_{ij} b_j^{(1)}$ is replaced by the canonical distribution $D_{\mathcal{I},\sigma_1^*(B^{(1)})^T}$, so $u_i = [(e_{i,L} + w_i + c_i) \cdot z^{-1}]_q$, with $w_i \hookleftarrow D_{\mathcal{I},\sigma_1^*(B^{(1)})^T}$ for $0 \le i \le \kappa$.
- Game_3: Modification of Game_2 in which offset vector c_i in the randomization of encoding u_i is removed and replaced by $-e_{i,L}$, so that $u_i = [(e_{i,L} + w_i) \cdot z^{-1}]_q$, where $w_i \hookleftarrow D_{\mathcal{I},\sigma_1^*(B^{(1)})^T,-e_{i,L}}$ for $0 \le i \le \kappa$ (note that $e_{i,L} + w_i$ is distributed as $D_{\mathcal{I}+e_{i,L},\sigma_1^*(B^{(1)})^T}$ over the randomness of w_i).
- Game_4: Modification of Game_3 in which e_i is sampled from $D_{R,\sigma'}$ (instead of sampling e_i from the truncated tail Gaussian $D^t_{R,\sigma'}$), for $0 \le i \le \kappa$, and a is sampled from $D_{1+\mathcal{I},\sigma'}$ (instead of $D^t_{1+\mathcal{I},\sigma'}$).
- Game_5: The Ext-cGCDH experiment, which can be obtained as a modification of Game_4 in which $e_{i,L}$ is sampled uniformly from R_g, instead of being computed from e_i as $e_{i,L} = [e_i]_g$.

For $i = 0, \ldots, 5$, let V_i denote the distribution of the view of A in Game_i, and let E denote the event that A outputs the correct Ext-GCDH solution. By the probability preservation property of RD from Lemma 4.1, we have that the advantage of A against Ext-cGCDH is $V_5(E) \ge V_1(E)^2 / R(V_1 \| V_5)$ and from the probability preservation property of the SD, the latter is $\ge (\varepsilon - \Delta(V_0, V_1))^2 / R(V_1 \| V_5)$.

To complete the proof, it thus remains to show that $\Delta(V_0, V_1) = O(\kappa \cdot 2^{-n})$ and $R(V_1 \| V_5) \le R$, with R defined in the theorem statement. Using two applications of the weak triangle inequality and one application of the R_∞ triangle inequality from Lemma 4.1, we get $R(V_1 \| V_5) \le R_\infty(V_1 \| V_2)^2 \cdot R(V_2 \| V_5)$, $R(V_2 \| V_5) \le R(V_2 \| V_3) \cdot R_\infty(V_3 \| V_5)$ and finally

$$R(V_1 \| V_5) \le R_\infty(V_1 \| V_2)^2 \cdot R(V_2 \| V_3) \cdot R_\infty(V_3 \| V_4) \cdot R_\infty(V_4 \| V_5).$$

We now bound each factor in turn:

- To bound $\Delta(V_0, V_1)$, we use the fact that Game_0 and Game_1 differ only if the norm of one of the sampled e_i (for $i \in \{0, \ldots, \kappa\}$) or a exceeds $2\sqrt{n} \cdot \sigma'$. By Lemma 2.3, since $\sigma' \ge \eta_{1/2}(\mathcal{I})$ (which follows from the samp uniformity smoothing condition, as shown below), this event occurs with probability at most 2^{-n+2} for each of these $\kappa + 2$ Gaussian samples. By the union bound, it thus follows that
$$\Delta(V_0, V_1) \le (\kappa + 2) \cdot 2^{-n+2} = O(\kappa \cdot 2^{-n}).$$
- To bound $R_\infty(V_1 \| V_2)^2$, we apply our LHL over R (Theorem 5.1) to conclude that, for each $i \in [\kappa + 1]$, $R_\infty(D(\sum_j \rho_{ij} b_j^{(1)}) \| D_{\mathcal{I},\sigma_1^*(B^{(1)})^T}) \le 1 + 4\varepsilon_\rho \le \exp(4\varepsilon_\rho)$ if $\varepsilon_\rho \le 1/2$, $\sigma_1^* \ge \|g^{-1} B^{(1)}\|_\infty n \sqrt{2 \log(4n \cdot \varepsilon_\rho^{-1})/\pi}$, and $B^{(1)} \cdot R^2 = \mathcal{I}$. The last condition on $B^{(1)}$ holds by the rejection test of the $\mathsf{InstGen}$ algorithm of $\mathsf{GGHLite}$. The condition on σ_1^* holds by the assumed LHL Smoothing condition and the bound $\|g^{-1} \cdot B^{(1)}\|_\infty \le \|g^{-1}\| \cdot \|B^{(1)}\| \le \ell_{g^{-1}} \cdot \sigma' \cdot \sqrt{n}$, from the rejection tests of the $\mathsf{InstGen}$ algorithm. Using the multiplicativity property over $i \in [\kappa + 1]$, and data processing inequality for R_∞, we conclude that
$$R_\infty(V_1 \| V_2)^2 \le \exp(8 \cdot (\kappa + 1) \cdot \varepsilon_\rho).$$

– To bound $R(V_2\|V_3)$, let $D_{1,i} = D_{\mathcal{I},\sigma_1^*(B^{(1)})^T} + c_i = D_{\mathcal{I},\sigma_1^*(B^{(1)})^T,c_i}$ (using $\mathcal{I} + c_i = \mathcal{I}$, since $c_i \in \mathcal{I}$) and $D_{2,i} = D_{\mathcal{I},\sigma_1^*(B^{(1)})^T,-e_{i,L}}$ for $i \in [\kappa+1]$. From the offset drowning condition, we have $\sigma_1^* \cdot \ell_b \geq \sigma'$, and using the samp uniformity smoothing condition, we have $\sigma' \geq \eta_{\varepsilon_e}(\mathcal{I})$, where we have used the bound $\eta_{\varepsilon_e}(\mathcal{I}) \leq \sqrt{\frac{\ln(2n(1+1/\varepsilon_e))}{\pi}} \cdot \lambda_n(\mathcal{I})$ from Lemma 2.4, and the fact that $\lambda_n(\mathcal{I}) = \lambda_1(\mathcal{I}) \leq \|g\| \leq \sqrt{n} \cdot \sigma$. We conclude that $\sigma_n(\sigma_1^*(B^{(1)})^T) \geq \sigma_1^* \cdot \ell_b \geq \eta_{\varepsilon_e}(\mathcal{I})$. Therefore, we can apply our offset Gaussian divergence bound (Lemma 4.2) for each i (with $w = c_i$ and $z = -e_{i,L}$) to get that, conditioned on a fixed value of offset c_i and encoded element $e_{i,L}$ (as well as fixed g, $B^{(1)}$ and a), we have $R(D_{1,i}\|D_{2,i}) \leq (\frac{1+\varepsilon_e}{1-\varepsilon_e})^2 \cdot \exp(2\pi\|c_i + e_{i,L}\|^2/(\sigma_1^*\sigma_n(B^{(1)}))^2) \leq \exp(2\pi\|c_i+e_{i,L}\|^2/(\sigma_1^*\ell_b)^2 + 8\varepsilon_e)$ using $(\frac{1+\varepsilon_e}{1-\varepsilon_e})^2 \leq \exp(8\varepsilon_e)$ for $\varepsilon_e < 1/2$. We also have $\|c_i + e_{i,L}\| = \|e_i \cdot a\| \leq \sqrt{n} \cdot \|e_i\| \cdot \|a\| \leq n^{1.5} \cdot (\sigma')^2$, using the bounds $\|e_i\| \leq \sqrt{n} \cdot \sigma'$, $\|a\| \leq 2\sqrt{n}\sigma'$. Therefore, we get $R(D_{1,i}\|D_{2,i}) \leq \exp(\varepsilon_d + 8\varepsilon_e)$ using the "Offset Drowning" condition. Using the multiplicativity property over $i \in [\kappa+1]$, and data processing property of R, we conclude that

$$R(V_2\|V_3) \leq \exp((\kappa+1) \cdot (\varepsilon_d + 8\varepsilon_e)).$$

– To bound $R_\infty(V_3\|V_4)$, we recall that for each $i \in [\kappa+1]$, the distribution $D^t_{R,\sigma'}$ of e_i in Game_3 is obtained by rejecting and resampling from $D_{R,\sigma}$ if the rejection test $\|e_i\| > \sqrt{n}\sigma'$ is satisfied. It follows that $D^t_{R,\sigma'}(x) = \frac{1}{1-p_{rej}} \cdot D_{R,\sigma'}(x)$ for all x in the support of $D^t_{R,\sigma'}$, where p_{rej} is the probability that a sample $D_{R,\sigma'}$ is rejected, and hence that $R_\infty(D^t_{R,\sigma'}\|D_{R,\sigma'}) = \frac{1}{1-p_{rej}}$. By the discrete Gaussian tail bound Lemma 2.3, we have $p_{rej} \leq 2^{-n+2}$ if $\sigma' \geq \eta_{1/2}(R)$, and the latter condition is satisfied by the choice of σ'. It follows that $R_\infty(D^t_{R,\sigma'}\|D_{R,\sigma'}) \leq 1 + 2^{-n+3}$. Applying a similar argument to the distribution of a using $\sigma' \geq \eta_{1/2}(\mathcal{I})$, we have $R_\infty(D^t_{1+\mathcal{I},\sigma'}\|D_{1+\mathcal{I},\sigma'}) \leq 1 + 2^{-n+3}$ and hence by the multiplicativity and data processing properties of the RD:

$$R_\infty(V_3\|V_4) \leq (1 + 2^{-n+3})^{\kappa+2} \leq \exp((\kappa+2) \cdot 2^{-n+3}).$$

– To bound $R_\infty(V_4\|V_5)$, let D_e denote the distribution of $[e_i]_g$ over the randomness of e_i sampled from $D_{R,\sigma'}$. We apply smoothing Lemma 2.6. to get that $R_\infty(U(R_g)\|D_e) \leq \frac{1+\varepsilon_e}{1-\varepsilon_e}$ if $\sigma' \geq \eta_{\varepsilon_e}(I)$. The latter condition holds as shown above. Using the multiplicativity and data processing properties of RD from Lemma 4.1, over $i = 0, \ldots, \kappa$, we conclude that for $\varepsilon_e \leq 1/2$:

$$R_\infty(V_4\|V_5) \leq \left(\frac{1+\varepsilon_e}{1-\varepsilon_e}\right)^{\kappa+1} \leq \exp((\kappa+1) \cdot 4\varepsilon_e).$$

Combining the above bounds gives the claimed bound. For the last statement, it suffices to observe that $\varepsilon' = \Omega(\varepsilon^2/\mathrm{poly}(\lambda))$ if $\kappa \cdot \max(\varepsilon_d, \varepsilon_\rho, \varepsilon_e) = O(\log \lambda)$. □

6.1 Canonical re-randomization algorithm cenc.

In Remark 2 of [10], the authors of the original GGH scheme define a canonicalizing encoding algorithm cenc that allows for certain applications (like the ABE scheme in [12]) to use the encoding re-randomization multiple times. We can define such a canonical re-randomization algorithm for our $\mathsf{GGHLite}$ in a similar way.

Algorithm $\mathsf{cenc}_l(\mathrm{par}, k, u')$ takes a level-k encoding u' of some element $e \in R_g$ with $k \leq \kappa$ and returns a re-randomized level-k encoding u of e. The parameter l indicates the "re-randomization depth," i.e., the number of times that cenc has been applied, and determines the re-randomization noise level.

Alternative "pairwise closeness" re-randomization security requirement. For applications such as the ABE scheme in [12], it is required that, for any two given level-k encodings $u'_1 = [c_1/z^k]_q, u'_2 = [c_2/z^k]_q$ of the same element e, the pair of distributions $D(u_1), D(u_2)$ of $u_1 = \mathsf{cenc}_l(\mathrm{par}, k, u'_1)$ and $u_2 = \mathsf{cenc}_l(\mathrm{par}, k, u'_2)$, respectively (over the randomness of cenc), are "close." This "pairwise closeness" requirement for re-randomized encodings is an alternative to the "closeness to a canonical distribution" requirement for re-randomized encodings in Definition 3.2 and Definition 6.1. In the case of the strong SD-based "closeness" requirement in Definition 3.2, we have, from the triangle inequality property of SD, that the "closeness to a canonical distribution" requirement of Definition 3.2 implies the "pairwise closeness" requirement. However, due to the lack of such a general triangle inequality property for the RD, such an implication does not immediately hold for our weak RD-based "closeness" requirements. Nevertheless, our improved re-randomization analysis of $\mathsf{GGHLite}$ above can be carried over to establish the weak "pairwise closeness" requirement as well.

In the following, we define our weak RD-based "pairwise closeness" re-randomization requirement.

Definition 6.3 (Weak pairwise-closeness re-randomization property of cenc). *Fix a κ-graded encoding scheme S, and an instance* par *of this scheme for security parameter λ. For $k \leq \kappa$ and $l \leq L$, let $S_{(k,l)}$ denote a set of "admissible" level-k input encodings at re-randomization depth l. Let cenc_l denote a re-randomization probabilistic algorithm that takes as input (par, k, u') with u' a level-k encoding of some level-0 element e_L, and returns a re-randomized level-k encoding u of e_L. Then we say that cenc satisfies the* weak pairwise closeness re-randomization *property for S with RD bound R and admissible input encoding sets $\{S_{(k,l)}\}_{k\in[\kappa],l\in[L]}$ if, for any $k \in [\kappa]$, $l \in [L]$ and two level-k encodings $u'_1, u'_2 \in S_{(k,l)}$ of the same level 0 element e_L, we have $R(D(u_1)||D(u_2)) \leq R = O(\mathrm{poly}(\lambda))$, where $D(u_i)$ denotes the distribution (over the randomness of cenc) of the re-randomized encoding $u_i = \mathsf{cenc}_l(\mathrm{par}, k, u'_i)$ for $i \in \{1, 2\}$.*

Next, we show that our requirement above is satisfied for $\mathsf{GGHLite}$ by a canonical re-randomization algorithm cenc with a similar choice of parameters as in Theorem 6.2. The proof is very similar to the proof of Theorem 6.2. The main difference is the direct "jump" in the RD-based analysis between the pair of encoding distributions $D(u_1), D(u_2)$ to avoid going through an intermediate canonical distribution, which would require applying a "strong" triangle inequality for the RD.

Lemma 6.4 (Weak Pairwise-closeness Re-randomization for $\mathsf{GGHLite}$). *Let $\varepsilon_d, \varepsilon_\rho, \varepsilon_e \in (0, 1/2)$ and $\kappa \leq 2^n$. For $k \leq \kappa$ and $l \in [L]$, let $\mathsf{cenc}_l(\mathrm{par}, k, u')$ denote the canonicalizing encoding algorithm for $\mathsf{GGHLite}$ that takes a level-k encoding $u' = [c'/z^k]_q$ with $\|c'\| \leq \gamma_{k,l}$, and returns a re-randomized encoding $u = [u' + \rho_1 \cdot x_1^{(k)} + \rho_2 \cdot x_2^{(k)}]_q$ with $\rho_1, \rho_2 \hookleftarrow D_{R,\sigma^*_{k,l}}$, for some admissible input encoding norm bound $\gamma_{k,l}$. Suppose that the following conditions hold:*

- ***LHL Smoothing:***
$$\sigma^*_{k,l} \geq n^{1.5} \cdot \ell_{g^{-1}} \cdot \sigma' \cdot \sqrt{2\log(4n \cdot \varepsilon_\rho^{-1})/\pi}. \tag{12}$$
- ***Offset "Drowning:"***
$$\sigma^*_{k,l} \geq (\sqrt{8\pi\varepsilon_d^{-1}}/\ell_b) \cdot \gamma_{k,l}. \tag{13}$$

Then cenc_l satisfies the weak pairwise-closeness re-randomization property for $\mathsf{GGHLite}$ with RD bound
$$R = \exp(12\varepsilon_\rho + \varepsilon_d), \tag{14}$$
and admissible input encoding sets $S_{k,l} = \{u' = [c'/z^k]_q : \|c'\| \leq \gamma_{k,l}\}$.

Proof. We fix an instance $\mathrm{par} = (n, q, y, \{(x_1^{(k)}, x_2^{(k)})\}_{k\leq\kappa})$ and p_{zt} of $\mathsf{GGHLite}$, with $x_1^{(k)} = [b_1^{(k)}/z^k]_q$, $x_2^{(k)} = [b_2^{(k)}/z^k]_q$, and $y = [a/z]_q$ with $a = 1+gr_y$, and two level-k encodings $u_i' = [c_i'/z^k]_q$ in $S_{k,l}$, i.e. with $\|c_i'\| \leq \gamma_{k,l}$, of the same level 0 element e_L, so that $c_i' = e_L + c_i \in R$ with $c_i \in \mathcal{I}$ for $i \in \{1, 2\}$. We consider the following sequence of games, where in each game, a re-randomized level-k encoding u of e_L is sampled, but the distribution of u differs among the games as follows:

- Game_0: In this game, we define u as the re-randomization of u_1', i.e. $u = \mathsf{cenc}_l(\mathrm{par}, k, u_1') = [(e_L + c_1 + w)/z^k]_q$, where $w = \rho_1 \cdot b_1^{(k)} + \rho_2 \cdot b_2^{(k)} \in R$ and $\rho_i \hookleftarrow D_{R,\sigma_{k,l}^*}$ for $i \in \{1, 2\}$.
- Game_1: Modification of Game_1 in which the distribution of the re-randomization term w is replaced by the distribution $D_{\mathcal{I},\sigma_{k,l}^*(B^{(k)})^T}$, i.e. $u = [(e_L + c_1 + w)/z^k]_q$ with $w \hookleftarrow D_{\mathcal{I},\sigma_{k,l}^*(B^{(k)})^T}$.
- Game_2: Modification of Game_2 in which the randomization offset term $c_1 \in \mathcal{I}$ is replaced by offset term $c_2 \in \mathcal{I}$, i.e. $u = [(e_L + c_2 + w)/z^k]_q$ with $w \hookleftarrow D_{\mathcal{I},\sigma_{k,l}^*(B^{(k)})^T}$.
- Game_3: Modification of Game_2 which "undoes" the modification introduced in Game_1, i.e. in this game we have $u = [(e_L + c_2 + w)/z^k]_q$, where $w = \rho_1 \cdot b_1^{(k)} + \rho_2 \cdot b_2^{(k)} \in R$ and $\rho_i \hookleftarrow D_{R,\sigma_{k,l}^*}$ for $i \in \{1, 2\}$. Observe that in this game, u has exactly the distribution of a re-randomization of u_2', i.e. $u = \mathsf{cenc}_l(\mathrm{par}, k, u_2')$.

For $i = 0, \ldots, 3$, let $D(u)_i$ denote the distribution of u in Game_i. To prove the lemma, it thus suffices to show that $R(D(u)_0\|D(u)_3) \leq R$, with R defined in the lemma statement. Applying both of the weak triangle inequalities from Lemma 4.1, we get

$$R(D(u)_0\|D(u)_3) \leq R_\infty(D(u)_0\|D(u)_1)^2 \cdot R(D(u)_1\|D(u)_2) \cdot R_\infty(D(u)_2\|D(u)_3).$$

We now bound each factor in turn:

- To bound $R_\infty(D(u)_0\|D(u)_1)^2$, we apply our LHL over R (Theorem 5.1) to conclude that $R_\infty(D(u)_0\|D(u)_1) \leq 1 + 4\varepsilon_\rho$ if $\sigma_{k,l}^* \geq \|g^{-1}B^{(1)}\|_\infty n\sqrt{2\log(4n \cdot \varepsilon_\rho^{-1})/\pi}$, and $B^{(1)} \cdot R^2 = \mathcal{I}$. The last condition on $B^{(1)}$ holds by the rejection test of the $\mathsf{InstGen}$ algorithm of $\mathsf{GGHLite}$. The condition on $\sigma_{k,l}^*$ holds by the assumed LHL Smoothing condition and the bound $\|g^{-1} \cdot B^{(k)}\|_\infty \leq \|g^{-1}\| \cdot \|B^{(k)}\| \leq \ell_{g^{-1}} \cdot \sigma' \cdot \sqrt{n}$, from the rejection tests of the $\mathsf{InstGen}$ algorithm. Using the data processing inequality for R_∞, we conclude that
$$R_\infty(D(u)_0\|D(u)_1)^2 \leq \exp(8\varepsilon_\rho).$$
- To bound $R(D(u)_1\|D(u)_2)$, notice that for $i \in \{1, 2\}$, using the fact that $c_i \in \mathcal{I}$, the distribution of $c_i + w$ in Game_i is $D_i \stackrel{\text{def}}{=} D_{\mathcal{I},\sigma_{k,l}^*(B^{(k)})^T,c_i}$. Applying our offset Gaussian divergence bound (Lemma 4.2) (with $w = c_1, z = c_2$) gives $R(D(u)_1\|D(u)_2) \leq \exp(2\pi\|c_1 - c_2\|^2/(\sigma_{k,l}^*\sigma_n(B^{(k)}))^2)$. The latter is upper bounded by $\exp(\varepsilon_d)$ if $(\sigma_{k,l}^*)^2 \geq \frac{2\pi\|c_1-c_2\|^2}{\varepsilon_d \cdot \sigma_n(B^{(k)}))^2}$. This last condition is satisfied by the offset drowning condition, using $\|c_1-c_2\| = \|c_1'-c_2'\| \leq 2\gamma_{k,l}$ and the acceptance condition $\sigma_n(B^{(k)}) \geq \ell_b$ of the $\mathsf{InstGen}$ algorithm. We conclude that
$$R(D(u)_1\|D(u)_2) \leq \exp(\varepsilon_d).$$
- To bound $R_\infty(D(u)_2\|D(u)_3)$, we apply the LHL over R (Theorem 5.1) with the same argument as used to bound $R_\infty(D(u)_0\|D(u)_1)$, except that this time, the order of the arguments to R_∞ is reversed. Since the R_∞ upper bound of Theorem 5.1 holds regardless of the order, we conclude that
$$R_\infty(D(u)_2\|D(u)_3) \leq \exp(4\varepsilon_\rho).$$

Combining the above bounds gives the claimed bound R.

□

6.2 Eliminating z: an NTRU variant of GGHLite

In this section, we introduce a simplified variant of the GGH/GGHLite scheme that eliminates the parameter z, and yet preserves the security of the GDDH/GCDH problems. We call our variant the NTRU variant, since it involves publishing "NTRU-like" quotients $pk_i^{(k)} = [x_i^{(k)}/y^k]_q = [b_i^{(k)}/a^k]_q$ instead of the separate GGH parameters $x_i^{(k)}, y$, thus cancelling out the parameter z, and replacing it effectively by a. Similarly, level-k encodings in this construction also correspond to GGHLite encodings divided by y^k, i.e., have the form $u = [(e \cdot a^k + \rho_1 b_1^{(k)} + \rho_2 b_2^{(k)})/a^k]_q = [e + \rho_1 pk_1^{(k)} + \rho_2 pk_2^{(k)}]_q$. The zero testing parameter is accordingly modified to $p_{zt} = \frac{h}{g} a^k$. The latter encoding resembles an NTRU ciphertext for e with respect to public keys $pk_1^{(k)}, pk_2^{(k)}$, although in NTRU we have only one public key, whereas here we have two public keys. The fact that public parameters and encodings can be efficiently translated from GGHLite to the NTRU variant by taking quotients in R_q, implies that the security of the NTRU variant is at least as hard as GGHLite. Details of the scheme are summarized in Figure 5.

- **Instance Generation** $\mathsf{InstGen}(1^\lambda, 1^\kappa)$: Given security parameter λ and multilinearity parameter κ, determine scheme parameters n, q, $m_r = 2$, σ, σ', $\ell_{g^{-1}}$, ℓ_b,ℓ. Let $R = \mathbb{Z}[x]/(x^n + 1)$ and $R_q = R/qR = \mathbb{Z}_q[x]/(x^n + 1)$. Do the following:
  - Sample $g \hookleftarrow D^t_{R,\sigma}$. If (1) $\|g^{-1}\| > \ell_{g^{-1}}$ or (2) $\langle g \rangle$ is not a prime ideal, resample g, else define ideal $\mathcal{I} = \langle g \rangle$.
  - Sample $a \hookleftarrow D^t_{1+I,\sigma'}$ (note that $a = 1 + gr_y$ for some $r_y \in R$).
  - For $k \in [\kappa]$:
    - Sample $B^{(k)} = (b_1^{(k)}, b_2^{(k)})$ from $(D^t_{I,\sigma'})^2$. If: (1) $\langle b_1^{(k)}, b_2^{(k)} \rangle \neq \mathcal{I}$, or (2) $\sigma_n(\mathrm{rot}(B^{(k)})) < \ell_b$, resample.
    - Define level-k public keys: $pk_1^{(k)} = [b_1^{(k)} \cdot a^{-k}]_q$, $pk_2^{(k)} = [b_2^{(k)} \cdot a^{-k}]_q$.
  - Sample $h \hookleftarrow D_{R,\sqrt{q}}$ and define the zero-testing parameter: $p_{zt,\kappa} = [\frac{h}{g} a^\kappa]_q \in R_q$.
  - Return public parameters $\mathrm{par} = (n, q, \{(pk_1^{(k)}, pk_2^{(k)})\}_{k \in [\kappa]})$ and p_{zt}.
- **Level-k encoding** $\mathsf{enc}_k(\mathrm{par}, e)$: Given level-0 encoding $e \in R$ and parameters par, return $u = [e + \rho_1 \cdot pk_1^{(k)} + \rho_2 \cdot pk_2^{(k)})]_q$, with $\rho_1, \rho_2 \hookleftarrow D_{R,\sigma_k^*}$ (note $u = [(c' + \rho_1 b_1^{(k)} + \rho_2 b_2^{(k)})/a^k]_q$, where $c' = e \cdot a^k \in e + I$).

Fig. 5. The new algorithms of our NTRU variant GGHLite scheme. Other algorithms are the same as in the original GGH scheme.

Security of the construction. We can define the corresponding problems GCDH^{NTRU}, $\mathrm{ExtGCDH}^{NTRU}$ and GDDH^{NTRU} for this NTRU variant, in the natural way as in Section 3, but with respect to experiment of Figure 6.

To show that the NTRU variant of the GGH encoding scheme is at least as secure as the GGH scheme, we now provide a formal reduction from GDDH to GDDH^{NTRU} (and similarly for the other two problems).

Theorem 6.5. *There exists a polynomial time reduction from GDDH (resp. GCDH/ExtGCDH) to GDDH^{NTRU} (resp. GCDH^{NTRU}/$\mathrm{ExtGCDH}^{NTRU}$).*

Given parameters $\lambda, n, q, m_r, \kappa, \sigma'$, proceed as follows:

1. Run $\mathsf{InstGen}(1^n, 1^\kappa)$ to get $\text{par} = (n, q, \{pk_j^{(k)}\}_{j,k})$ and p_{zt}.
2. For $i = 0, \ldots, \kappa$:
   - Sample $e_i \hookleftarrow D_{R,\sigma'}$ and $f_i \hookleftarrow D_{R,\sigma'}$,
   - Set $u_i = [e_i + \sum_j \rho_{ij} pk_j]_q$ with $\rho_{ij} \hookleftarrow \chi_1$ for all j.
3. Set $u^* = \left[\prod_{i=1}^{\kappa} u_i\right]_q$.
4. Set $v_C = [e_0 u^*]_q$.
5. Sample $\rho_j \hookleftarrow \chi_\kappa$ for all j; set $v_D = [e_0 u^* + \sum_j \rho_j pk_j^{(\kappa)}]_q$.
6. Set $v_R = [f_0 u^* + \sum_j \rho_j pk_j^{(\kappa)}]_q$.

Fig. 6. The GGH^{NTRU} security experiment.

Proof. For simplicity, we only describe the reduction from GDDH to GDDH^{NTRU}. Let $\{(y, \{x_j\}_j, p_{zt}), u_0, \ldots, u_\kappa, v\}$ be a GDDH instance and let $\mathcal{O}$ be a polynomial-time oracle for solving GDDH^{NTRU}.

- Let $pk_j^{(k)} = [\frac{x_j^{(k)}}{y}]_q$ for $j \in \{1, 2\}$ and $k \in [\kappa]$,
- Let $\hat{p}_{zt} = [p_{zt} \cdot y^\kappa]_q$,
- Let $\hat{v} = [v \cdot y^{-\kappa}]_q$,
- Call the oracle $\mathcal{O}$ on input $\{(\{pk_j^{(k)}\}_{j,k}, \hat{p}_{zt}), [\frac{u_0}{y}]_q, \ldots, [\frac{u_\kappa}{y}]_q, \hat{v}\}$.

We have $u_i = \mathsf{enc}_1(e_i) = [e_i y + \sum_j \rho_j^{(i)} x_j^{(1)}]_q$ for all $i \in [\kappa]$, then let $u_i^{\text{NTRU}} = [\frac{u_i}{y}]_q = [e_i + \sum_j \rho_j^{(i)} \frac{x_j^{(1)}}{y}]_q = [e_i + \sum_j \rho_j^{(i)} pk_j^{(1)}]_q$ is a valid NTRU variant level-1 encoding for e_i. Furthermore, if $v = v_D$, then

$$\hat{v} = [(e_0 \cdot u^* + \sum_j \rho_j \cdot x_j^{(\kappa)}) \cdot y^{-\kappa}]_q = [e_0 \cdot \prod_{i=1}^{\kappa} (\frac{u_i}{y}) + \sum_j \rho_j \cdot \frac{x_j^{(\kappa)}}{y^\kappa}]_q = [e_0 \cdot \prod_{i=1}^{\kappa} u_i^{\text{NTRU}} + \sum_j \rho_j \cdot pk_j^{(\kappa)}]_q,$$

is a valid NTRU variant level-κ encoding of $\prod_i e_i$. Similarly, if $v = v_R$, then $\hat{v}$ is a valid NTRU variant level-κ encoding of $f_0 \prod_{i \geq 1} e_i$, as required. □

7 Parameter settings

In Table 1, we summarize asymptotic parameters for GGHLite to achieve 2^λ security for the underlying Ext-GCDH problem, assuming the hardness of the canonical Ext-cGCDH problem, and to satisfy the zero-testing/extraction correctness conditions with error probability $\lambda^{-\omega(1)}$. For simplicity, we assume that $\kappa = \omega(1)$ and $\kappa = O(\text{poly}(\lambda))$. For comparison, we also show the corresponding parameters for GGH. The "Condition" column lists the conditions that determine the corresponding parameter in the case of GGHLite. For security of the canonical Ext-cGCDH problem, we assume (as in [10]) that the best attack is the one described in [10, Se. 6.3.3], whose complexity is dominated by the cost of solving γ-SVP (the Shortest lattice Vector Problem with approximation factor γ) for

the lattice $\mathcal{I}$, with γ set at $\approx q^{3/8}$ to get a sufficiently short multiple of g. By the lattice reduction "rule of thumb," to make this cost 2^λ, we need to set

$$n = \Omega(\lambda \log q). \tag{15}$$

Table 1. Asymptotic parameters.

Parameter	GGHLite	GGH[10]	Condition
m_r	2	$\Omega(n \log n)$	LHL: Th. 5.1
σ	$O(n \log n)$	$O(n \log n)$	Eq. (4)
$\ell_{g^{-1}}$	$O(1/\sqrt{n \log n})$	$O(1/\sqrt{n \log n})$	Eq. (4)
$\varepsilon_d, \varepsilon_e, \varepsilon_\rho$	$O(\kappa^{-1})$	$O(2^{-\lambda}\kappa^{-1})$	Eq. (11)
σ'	$\widetilde{O}(n^{3.5})$	$\widetilde{O}(n^{1.5}\sqrt{\lambda})$	Eq. (6)
σ_1^*	$\widetilde{O}(n^{5.5}\sqrt{\kappa})$	$\widetilde{O}(2^\lambda \lambda n^{4.5}\kappa)$	Drown: Eq. (9)
ε_{ext}	$O(\lambda^{-\omega(1)})$	$O(\lambda^{-\omega(1)})$	
q	$\widetilde{O}(n^{10.5}\sqrt{\kappa})^{8\kappa}$	$\widetilde{O}(2^\lambda \lambda^{1.5} n^{8.5}\kappa)^{8\kappa}$	Corr.: Eq. (7)
n	$O(\kappa\lambda \log \lambda)$	$O(\kappa\lambda^2)$	SVP: Eq. (15)
$\lvert\mathsf{enc}\rvert$	$O(\kappa^2 \lambda \log^2 \lambda)$	$O(\kappa^2\lambda^3)$	$O(n \log q)$
$\lvert\mathrm{par}\rvert$	$O(\kappa^3 \lambda \log^2 \lambda)$	$O(\kappa^4\lambda^5 \log \lambda)$	$O(m_r \kappa n \log q)$

When $\kappa = \mathrm{poly}(\log \lambda)$, the dimension n, encoding length $|\mathsf{enc}|$ and public parameters length $|\mathrm{par}|$ in our scheme GGHLite are all asymptotically close to optimal, namely quasi-linear in the security parameter λ, versus quadratic (resp. cubic and quintic) in λ for GGH [10]. Thus we expect GGHLite's public parameters and encodings to be orders of magnitudes shorter than GGH for typical $\lambda \approx 100$.

8 Applications

In previous sections, we have shown that our graded encoding scheme GGHLite can be instantiated much more efficiently than the GGH scheme [10], but on the other hand, with our efficient choice of parameters for GGHLite, we have only been able prove the hardness of the *search* problem Ext-GCDH (based on the hardness of the corresponding canonical problem) rather than the *decision* problem GDDH used in [10]. In this section, we show that the hardness of Ext-GCDH is sufficient for important applications of graded encoding schemes, in the random oracle model. In particular, we show that existing protocols based on the hardness of GDDH can be easily modified to make their security based on Ext-GCDH in the random oracle model, while preserving the efficiency of the original protocols, up to a small factor.

8.1 Efficient one-round N-party Diffie-Hellman key exchange in the ROM

We show how to adapt the one round N-party key exchange protocol described in [10, Section 5] (originally described by Boneh and Silverman [7] in the abstract setting of multilinear maps) to achieve security assuming the hardness the Ext-GCDH problem, rather than the GDDH problem, in the random oracle model. The modification is straightforward: we simply replace the shared key $s = \mathsf{ext}(par, p_{zt}, v)$ in the original protocol, where v is the encoding of the Diffie-Hellman product of the N parties' secrets, by its hash $K = H(\mathsf{ext}(par, p_{zt}, v))$, where $H(.) : \{0,1\}^* \to \{0,1\}^\lambda$ denotes a hash function modelled as a random oracle. Details follow.

Construction. Given a κ-graded encoding scheme with $\kappa = N-1$ over an encoded element ring $R/\mathcal{I}$ of prime order p, and a hash function $H : \{0,1\}^* \to \{0,1\}^\lambda$, the N-party key exchange protocol is presented in Figure 7.

- **Setup** $\mathsf{Setup}(1^\lambda, 1^N)$: Given security parameter λ and number of parties N, run $\mathsf{InstGen}(1^{2\lambda+1}, 1^{N-1})$ for the graded encoding scheme to get (par, p_{zt}) and output protocol public parameters (par, p_{zt}).
- **Publish** $\mathsf{Publish}(\mathrm{par}, p_{zt}, i)$: The ith party runs the level-0 encoding sampler to generate a random secret key $e_i = \mathsf{Samp}(\mathrm{par})$ (corresponding to encoded element $e_{i,L}$), and publishes a corresponding level-1 public key $u_i = \mathsf{enc}_1(\mathrm{par}, e_i)$.
- **KeyGen** $\mathsf{KeyGen}(\mathrm{par}, p_{zt}, j, e_j, \{u_i\}_{i\neq j})$: The jth party computes a level-$(N-1)$ encoding $v_j = e_j \cdot \prod_{i\neq j} u_i$ of the Diffie-Hellman product $\prod_i e_{i,L}$, and computes the key $K_j = H(s_j)$, where $s_j = \mathsf{ext}(\mathrm{par}, p_{zt}, v_j)$ is the extracted string for v_j.

Fig. 7. Our modified N-party Diffie-Hellman key exchange protocol.

Correctness. We have to show that all the N computed keys $K_1, \ldots, K_N$ are equal except for negligible probability $\lambda^{-\omega(1)}$. In the KeyGen algorithm, each party computes an encoding v_j of the product $e_L = \prod_i e_{i,L}$ in the ring $R/\mathcal{I}$. Since $|R/\mathcal{I}| = \Omega(2^\lambda)$ is prime and the distribution of the $e_{i,L}$'s is within statistical distance $O(2^{-\lambda})$ of uniform on $R/\mathcal{I}$, the product e_L is also within negligible statistical distance $O(2^{-\lambda})$ to a uniformly random element in $R/\mathcal{I}$. Hence by the extraction correctness property of the encoding scheme, all N extracted strings $\{s_j\}_{i\in[N]}$, and hence also all N computed keys $\{K_j\}_{i\in[N]}$, are equal, except with negligible probability $O(N \cdot \lambda^{-\omega(1)}) = O(\lambda^{-\omega(1)})$ for $N = \lambda^{O(1)}$.

Passive security. We have to show that, given (par, p_{zt}) and the public keys $u_1, \ldots, u_N$, the key (say K_1) is indistinguishable to the adversary $\mathcal{A}$ from a uniformly random string in $\{0,1\}^\lambda$, assuming the hardness of the Ext-GCDH problem and the random oracle model for H. Formally, we define a passive security attack game, in which $\mathcal{A}$ is given (par, p_{zt}), $u_1, \ldots, u_N$, and T_b, for a uniformly random bit $b \in \{0,1\}$, where $T_0 = K_1$ is the real key and $T_1 = R \leftarrow\!\!\!\!\!\!\rightarrow U(\{0,1\}^\lambda$ is an independent uniformly random string, and $\mathcal{A}$ outputs a guess b' for b. We say that $\mathcal{A}$'s advantage is $\varepsilon = 2(\Pr[b' = b] - 1/2)$.

Lemma 8.1. *Let $\mathcal{A}$ denote an attacker, in the random oracle model for H, against the passive security of the N-party Diffie-Hellman key exchange protocol in Figure 7, with run-time T and advantage ε, making q_H queries to H. Then there exists an algorithm $\mathcal{A}'$ for the Ext-GCDH problem for the underlying encoding scheme, with run-time $T' = T$ and success probability $\varepsilon' \geq \varepsilon/(2q_H)$.*

Proof. Let Game_1 denote the passive security attack game with $\mathcal{A}$, and let Game_2 denote a modification of Game_1 in which $\mathcal{A}$'s queries to H are answered differently as follows: if the query x is equal to $s_1 = \mathsf{ext}(\mathrm{par}, p_{zt}, e_1 \cdot \prod_{i>1} u_i)$, the query is answered with a uniformly random $K \in \{0,1\}^\lambda$ (instead of $K_1 = H(s_1)$), otherwise, the query is answered with $H(x)$, as in Game_1.

For $i \in \{1,2\}$, let S_i denote the event that $b' = b$ in Game_i, and let E denote the event in Game_1 that $\mathcal{A}$ queries H at s_1. Note that by definition, $\Pr[S_1] = 1/2 + \varepsilon/2$, and we also have $\Pr[S_2] = 1/2$ because in Game_2, T_b is a uniformly random string independent of $\mathcal{A}$'s prior view, regardless of the value of b. On the other hand, since the view of $\mathcal{A}$ is identical in Game_1 and Game_2 until $\mathcal{A}$ queries

H at s_1, we have $|\Pr[S_1] - \Pr[S_2]| \leq \Pr[E]$. It follows that $\Pr[E] \geq \varepsilon/2$. Given an input instance $(\mathrm{par}, p_{zt}, \{u_i\}_i)$ of the Ext-GCDH problem, the attacker $\mathcal{A}'$ simply runs $\mathcal{A}$ on input $(\mathrm{par}, p_{zt}, \{u_i\}_i)$ and T_b (with $T_0 = K_1$ chosen uniformly random in $\{0,1\}^\lambda$ – note that $\mathcal{A}'$ does not need to know s_1 to simulate T_0) and simulates Game_1, hoping that the event E occurs. Let $\{x_i\}_{i \in [q_H]}$ denote the queries made by $\mathcal{A}$ to H, When $\mathcal{A}$ finishes, $\mathcal{A}'$ chooses $i \in [q_H]$ uniformly at random and outputs x_i as its guess for $\mathcal{A}$'s query that equals s_1 (note that until $\mathcal{A}$ queries H at s_1, the view of $\mathcal{A}$ is perfectly simulated by $\mathcal{A}'$ as in Game_1, so $\Pr[E]$ is preserved). Conditioned on the event E occurring, we have $x_i = s_1$ with probability $\geq 1/q_H$. Overall, $\mathcal{A}'$ outputs the correct Ext-GCDH solution with probability $\geq 1/q_H \cdot \Pr[E] \geq \varepsilon/(2q_H)$. □

Note that when the protocol attacker $\mathcal{A}$ has run-time $T = 2^\lambda$ (so that also $q_H \leq 2^\lambda$) and advantage $\varepsilon \geq 2^{-\lambda}$, the Ext-GCDH attacker $\mathcal{A}'$ constructed by our security lemma above, has run-time $T' = 2^\lambda$ and advantage $\varepsilon' \geq 2^{-(2\lambda+1)}$, thus contradicting the assumed $2^{2\lambda+1}$-security of the undrlying encoding scheme (it is for this reason that we used a security parameter $\lambda' = 2\lambda+1$ for the encoding scheme). Consequently, we only lose a constant factor ≈ 2 in relating the security parameter of the encoding scheme to that of the protocol, essentially preserving the efficiency of our encoding scheme in this application.

8.2 Efficient ABE from canonical Ext-GCDH in the ROM

We explain how to modify the Attribute Based Encryption (ABE) scheme for circuits by Garg et al. [12] and its security proof, to achieve security assuming the hardness the Ext-GCDH problem in the random oracle model, and our *weak* canonical re-randomization property from Lemma 6.4, rather than the GDDH problem and the *strong* canonical re-randomization property from [10], although in this application, we need to set the re-randomization bound $R(D(u_1), D(u_2)) = \exp(O(\varepsilon_d + \varepsilon_\rho))$ to be relatively small, namely $O(1/Nq_f)$, where Nq_f is the total size of the circuits queried to the key generation oracle by the adversary. This still gives our scheme significant savings when Nq_f is much smaller than 2^λ. Our modification of the scheme uses the same hashed-key idea as used in the key-exchange protocol of the previous section. Since the scheme and its analysis are almost identical to that in [12], we only summarize the required changes below and refer the reader to [12] for details.

Construction. Given a κ-graded encoding scheme over an encoded element ring $R/\mathcal{I}$ of prime order p with canonical re-randomization algorithm cenc (see end of Section 6) and a hash function $G : \{0,1\}^* \to \{0,1\}$, the ABE scheme is presented in Figure 8.

Correctness. The correctness analysis in [12] shows that $\mathsf{Decrypt}$ correctly recovers a level-κ encoding E of $\alpha \cdot s$ if $f(x) = 1$. By the extraction property of the encoding scheme and the random choice of s, we have $v' = \mathsf{ext}(\mathrm{par}, p_{zt}, E)$ is equal to $v = \mathsf{ext}(\mathrm{par}, p_{zt}, H \cdot s)$ and therefore decryption succeeds to recover M, except with negligible probability $O(\lambda^{-\omega(1)})$.

Security. We sketch how to modify the security proof of selective security from [12, Theorem 6.1]. Full security follows as in [12, Corollary 6.2]. The selective security game consists of the following game. In the Init. stage, the adversary commits to the challenge attribute string x^*. Then the challenger runs Setup, gives PP to the adversary and keeps SK to itself. The adversary then makes q_f private key queries f of his choice such that $f(x^*) = 0$ to get keys $\mathsf{KeyGen}(MSK, f)$. The

- **Setup** $\mathsf{Setup}(1^\lambda, 1^n, \ell_c)$: Given security parameter λ, maximum circuit depth ℓ_c and number of circuit inputs n, run $\mathsf{InstGen}(1^{2\lambda+1}, 1^{\kappa=\ell_c+1})$ for the graded encoding scheme to get (par, p_{zt}). Sample $\alpha, \hat{h}_1, \ldots, \hat{h}_n$ using $\mathsf{Samp}(\mathrm{par})$. Return public parameters $PP = (\mathrm{par}, p_{zt}, H = \mathsf{cenc}_2(\mathrm{par}, k, \mathsf{enc}_k(\mathrm{par}, \alpha)), \{h_i = \mathsf{cenc}_2(\mathrm{par}, 1, \mathsf{enc}_1(\mathrm{par}, \hat{h}_i))\}_{i\in[n]})$ and Master secret key $MSK = \alpha$.
- **Encrypt** $\mathsf{Encrypt}(PP, x \in \{0,1\}^n, M \in \{0,1\})$: Sample $s = \mathsf{Samp}(\mathrm{par})$, and compute the key $K = G(v)$, where $v = \mathsf{ext}(\mathrm{par}, p_{zt}, H \cdot s)$ is the extracted string for the encoding $H \cdot s$ of $\alpha \cdot s$, and compute $C_M = M \oplus K \in \{0,1\}$. Let S denote the set of i such that $x_i = 1$. Return the ciphertext $C = (C_M, \hat{s} = \mathsf{cenc}_1(\mathrm{par}, 1, \mathsf{enc}_1(\mathrm{par}, s)), \{C_i = \mathsf{cenc}_3(\mathrm{par}, 1, h_i \cdot s)\}_{i\in S})$.
- **KeyGen** $\mathsf{KeyGen}(MSK = \alpha, f)$: Identical to [12]. Return SK consisting of function f, 'header' $K_H = \mathsf{cenc}_3(\mathrm{par}, \kappa - 1, \alpha - r_{n+q})$ (with $r_{n+q} \hookleftarrow \mathsf{Samp}(\mathrm{par})$), and key components $\{K_{w,i}\}_{w\in[n+q]}$.
- **Decrypt** $\mathsf{Decrypt}(SK, C)$: If $f(x) = 1$, compute a level-κ encoding E of $\alpha \cdot s$, as in [12] and recover $v' = \mathsf{ext}(\mathrm{par}, p_{zt}, E)$, $K' = G(v')$ and $M' = C_M \oplus K'$. Return message M'.

Fig. 8. Our modified ABE.

adversary then outputs a pair of messages M_0, M_1, and the challenger returns challenge ciphertext $C = \mathsf{Encrypt}(PP, x^*, M_b)$ for b a uniformly random bit. The challenger continues to run and outputs a guess b' for b. We say that $\mathcal{A}$'s advantage is $\varepsilon = 2(\Pr[b' = b] - 1/2)$.

Theorem 8.2. *Let $\mathcal{A}$ denote an attacker, in the random oracle model for G, against the selective security of the ABE scheme in Figure 8, with run-time T and advantage ε, making q_G queries to G and q_f private key queries on circuits f of $\leq N$ wires. Then there exists an algorithm $\mathcal{A}'$ for the Ext-GCDH problem for the underlying encoding scheme, with run-time $T' = T$ and success probability $\varepsilon' = \Omega(\varepsilon^{O(1)}/(q_G R^{O(Nq_f)}))$, where $R = R(D(u_1)\|D(u_2))$ is the canonical re-randomization RD bound for the canonical re-randomization algorithm* cenc *of the underlying encoding scheme (see Definition 6.3).*

Proof. (Sketch.) Let Game_1 denote the selective security attack game with $\mathcal{A}$, with $v = \mathsf{ext}(\mathrm{par}, p_{zt}, H \cdot s)$ being the extracted string of element αs, used to derive the key $K = G(v)$ in the challenge ciphertext C. As in the proof of Lemma 8.1, let Game_2 denote a modification of Game_1 in which $\mathcal{A}$'s queries to G are answered differently as follows: if the query x is equal to v, the query is answered with a uniformly random $K' \in \{0,1\}^\lambda$ (instead of $K = G(v)$), otherwise, the query is answered with $G(x)$, as in Game_1. Let E_i denote the event in Game_i that $\mathcal{A}$ queries G at v. As in the proof of Lemma 8.1, we have $\Pr[E_1] = \Pr[E_2] \geq \varepsilon/2$. Now, we define a sequence of games $\mathsf{Game}_3, \mathsf{Game}_4, \mathsf{Game}_5$, where in Game_5, algorithm $\mathcal{A}$ will solve the Ext-GCDH problem with respect to an input instance $\mathrm{par}, p_{zt}, \hat{s} = \mathsf{cenc}_1(\mathrm{par}, 1, \mathsf{enc}_1(\mathrm{par}, s))$, $\{\hat{c}_i = \mathsf{cenc}_1(1, c_i)\}_{i\in[\kappa]}$, where $s, c_1, \ldots, c_\kappa \hookleftarrow \mathsf{Samp}(\mathrm{par})$. Let V_i denote the view of $\mathcal{A}$ in Game_i.

- Game_3: Change the definition of h_i in PP to $h_i = \mathsf{cenc}_2(\mathrm{par}, 1, \mathsf{enc}_1(\mathrm{par}, y_i))$ if $x^*_i = 0$ and $h_i = \mathsf{cenc}_2(\mathrm{par}, 1, \mathsf{enc}_1(\mathrm{par}, y_i) + \hat{c}_1)$ if $x^*_i = 1$, where $y_i \hookleftarrow \mathsf{Samp}(\mathrm{par})$ for $i \in [n]$ (instead of $h_i = \mathsf{cenc}_2(\mathrm{par}, 1, \hat{h}_i)\}_{i\in[n]}$ in Game_2) . The encoded elements in the encodings that are input to cenc are uniformly random elements in $R/\mathcal{I}$ in both games, so by the weak canonical randomization and the multiplicativity property of RD over $i \in [n]$, we have $R(V_1\|V_3) \leq R^n$. By the probability preservation property of RD, it follows that $\Pr[E_3] \geq (\varepsilon/2)^2/R^n$.
- Game_4: Change the definition of H in PP to $H = \mathsf{cenc}_2(\mathrm{par}, k, \mathsf{enc}_k(\mathrm{par}, \xi) + \prod_{i\in[\kappa]} \hat{c}_i)$ (instead of $H = \mathsf{cenc}_2(\mathrm{par}, k, \mathsf{enc}_k(\mathrm{par}, \alpha))$), where $\xi \hookleftarrow \mathsf{Samp}(\mathrm{par})$ is random, effectively using element $\xi + \prod_i c_i$ to represent α. As above, the encoded element in H is a uniformly random element in $R/\mathcal{I}$ in both games, so by the weak canonical re-randomization assumption on cenc, we

have $R(V_3\|V_4) \leq R$. By the probability preservation property of RD, it follows that $\Pr[E_4] \geq \Pr[E_3]^2/R$.

– Game_5: Change the KeyGen query answers for circuits f with $f(x^*) = 0$ so that $\alpha = \xi + \prod_i c_i$ is not used explicitly in the computation of K_H. This change is described in the "KeyGen Phase" of the proof of [12, Theorem 6.1]. It involves changing the definition of key components $K_{w,i}$ for the wires w of f. The distribution of the encoded elements in the encodings re-randomized by cenc in the computation of $K_{w,i}$ are the same as in the previous game, but the input encodings have a different distribution. By the weak canonical re-randomization assumption on cenc and the multiplicativity property of RD over the $O(N \cdot q_f)$ key components in $\mathcal{A}$'s view, we have $R(V_4\|V_5) \leq R^{O(Nq_f)}$. By the probability preservation property of RD, it follows that $\Pr[E_5] \geq \Pr[E_4]^2/R^{O(Nq_f)}$.

In Game_5, algorithm $\mathcal{A}$ does not use the Ext-GCDH secrets $s, c_1, \ldots, c_\kappa$. The Ext-GCDH attacker $\mathcal{A}'$ simply runs $\mathcal{A}$ (with $K = H(v)$ chosen uniformly random in $\{0,1\}$ for computing the challenge ciphertext component C_M – note that $\mathcal{A}'$ does not need to know v to simulate C_M) and simulates Game_5, hoping that the event E_5 occurs. Let $\{x_i\}_{i\in[q_G]}$ denote the queries made by $\mathcal{A}$ to G. When $\mathcal{A}$ finishes, algorithm $\mathcal{A}'$ chooses $i \in [q_G]$ uniformly at random and outputs x_i as its guess for $\mathcal{A}$'s query that equals the Ext-GCDH solution v. Conditioned on the event E_5 occurring, we have $x_i = v$ with probability $\geq 1/q_G$. Overall, algorithm $\mathcal{A}'$ outputs the correct Ext-GCDH solution with probability $\geq 1/q_G \cdot \Pr[E_5] = \Omega(\varepsilon^{O(1)}/(q_G R^{O(Nq_f)}))$. □

Acknowledgments. We thank Vadim Lyubashevsky for useful discussions. This work has been supported in part by ERC Starting Grant ERC-2013-StG-335086-LATTAC, an Australian Research Fellowship (ARF) from the Australian Research Council (ARC), and ARC Discovery Grants DP0987734 and DP110100628.

References

1. D. Aggarwal and O. Regev. A note on discrete gaussian combinations of lattice vectors, 2013. Draft. Available at `http://arxiv.org/pdf/1308.2405v1.pdf`.
2. S. Agrawal, G. Gentry, S. Halevi, and A. Sahai. Discrete gaussian leftover hash lemma over infinite domains. In *Proc. of ASIACRYPT*, volume 8269 of *LNCS*, pages 97–116. Springer, 2013.
3. J. Alperin-Sheriff and C. Peikert. Circular and KDM security for identity-based encryption. In *Proc. of PKC*, volume 7293 of *LNCS*, pages 334–352. Springer, 2012.
4. G. Asharov, A. Jain, A. López-Alt, E. Tromer, V. Vaikuntanathan, and D. Wichs. Multiparty computation with low communication, computation and interaction via threshold FHE. In *Proc. of EUROCRYPT*, pages 483–501, 2012.
5. A. Banerjee, C. Peikert, and A. Rosen. Pseudorandom functions and lattices. In *Proc. of EUROCRYPT*, volume 7237 of *LNCS*, pages 719–737. Springer, 2012.
6. D. Boneh and M. K. Franklin. Identity-based encryption from the Weil pairing. *SIAM J. Comput.*, 32(3):586–615, 2003.
7. D. Boneh and A. Silverberg. Applications of multilinear forms to cryptography. *Contemporary Mathematics*, 324:71–90, 2003.
8. J.-S. Coron, T. Lepoint, and M. Tibouchi. Practical multilinear maps over the integers. In *Proc. of CRYPTO*, pages 476–493, 2013.
9. T. van Erven and P. Harremoës. Rényi divergence and Kullback-Leibler divergence. *CoRR*, abs/1206.2459, 2012.
10. S. Garg, C. Gentry, and S. Halevi. Candidate multilinear maps from ideal lattices. In *Proc. of EUROCRYPT*, volume 7881 of *LNCS*, pages 1–17. Springer, 2013.
11. S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *Proc. of FOCS*, pages 40–49. IEEE Computer Society Press, 2013.

12. S. Garg, C. Gentry, S. Halevi, A. Sahai, and B. Waters. Attribute-based encryption for circuits from multilinear maps. In *Proc. of CRYPTO*, volume 8043 of *LNCS*, pages 479–499. Springer, 2013.
13. C. Gentry. Fully homomorphic encryption using ideal lattices. In *Proc. of STOC*, pages 169–178. ACM, 2009.
14. C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proc. of STOC*, pages 197–206. ACM, 2008.
15. J. Hoffstein, J. Pipher, and J. H. Silverman. NTRU: A ring-based public key cryptosystem. In *Proc. of ANTS*, pages 267–288, 1998.
16. A. Joux. A one round protocol for tripartite Diffie-Hellman. In *Proc. of ANTS*, volume 1838 of *LNCS*, pages 385–394. Springer, 2000.
17. V. Lyubashevsky and D. Micciancio. Generalized compact knapsacks are collision resistant. In *Proc. of ICALP*, volume 4052 of *LNCS*, pages 144–155. Springer, 2006.
18. V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. *J. ACM*, 60(6):43, 2013.
19. D. Micciancio and S. Goldwasser. *Complexity of lattice problems: a cryptographic perspective*. Kluwer Academic Press, 2002.
20. D. Micciancio and O. Regev. Worst-case to average-case reductions based on Gaussian measures. *SIAM J. Comput*, 37(1):267–302, 2007.
21. P. Q. Nguyen and J. Stern. Merkle-Hellman revisited: A cryptanalysis of the Qu-Vanstone cryptosystem based on group factorizations. In *Proc. of CRYPTO*, volume 1294 of *LNCS*, pages 198–212. Springer, 1997.
22. C. Papamanthou, R. Tamassia, and N. Triandopoulos. Optimal authenticated data structures with multilinear forms. In *Proc. of Pairing*, pages 246–264, 2010.
23. O. Regev. Lecture notes of *lattices in computer science*, taught at the Computer Science Tel Aviv University. Available at `http://www.cims.nyu.edu/~regev/`.
24. A. Rényi. On measures of entropy and information. In *Proc. of the Fourth Berkeley Symposium on Math. Statistics and Probability*, volume 1, pages 547–561, 1961.
25. R. Rothblum. On the circular security of bit-encryption. In *Proc. of TCC*, pages 579–598, 2013.
26. M. Rückert and D. Schröder. Aggregate and verifiably encrypted signatures from multilinear maps without random oracles. In *Proc. of ISA*, pages 750–759, 2009.
27. R. Sakai, K. Ohgishi, and M. Kasahara. Cryptosystems based on pairing. *SCIS*, 2000.
28. D. Stehlé and R. Steinfeld. Making NTRU as secure as worst-case problems over ideal lattices. In *Proc. of EUROCRYPT*, volume 6632 of *LNCS*, pages 27–47. Springer, 2011. Conference version of [29].
29. D. Stehlé and R. Steinfeld. Making NTRUEncrypt and NTRUSign as secure standard worst-case problems over ideal lattices, 2013. Full version of [28], available at `http://perso.ens-lyon.fr/damien.stehle/NTRU.html`.

A Background on graded encoded schemes

We refer to [10] for the basic definitions concerning graded encoded schemes.

A.1 Correctness analysis of the GGH scheme

We explain here how to derive the correctness conditions of Section 3. For this, we need the following result.

Lemma A.1 (Adapted from [10, Lemma 4]). *Let $g \in R$ such that $\mathcal{I} = \langle g \rangle$ is a prime ideal in R, let $c \in R$ with $\|c\| < q^{1/8}$ and $h \in R$ with $\|h\| < \sqrt{n}q^{1/2}$ and $c, h \notin \mathcal{I}$ and $q > (2tn\sigma)^4$ for some $t \geq 1$. Then $\|[h \cdot c/g]_q\|_\infty > t \cdot q^{3/4}$.*

Correctness of zero-testing. To satisfy the "no false negatives" zero-testing condition, we need $\|p_{zt}u\|_\infty < q^{3/4}$ for all valid level-κ encodings $u = [c/z^\kappa]_q \in S_\kappa^{(0)}$ of zero. Taking $S_\kappa^{(0)}$ as the set of possible encodings obtained by multiplying κ level-1 encodings $u_i = [c_i/z]_q$ output by Enc, we have

$$\|[p_{zt}u]_q\|_\infty = \|[\frac{hc}{g}]_q\|_\infty = \|\frac{hc}{g}\|_\infty \leq \|hc\| \cdot \|g^{-1}\| \leq \|h\| \cdot \|c\| \cdot \|g^{-1}\|\sqrt{n}.$$

To satisfy $\|p_{zt}u\|_\infty < q^{3/4}$, it therefore suffices to have $\|c\| \le q^{1/8}$ and $\|h\| \cdot q^{1/8} \cdot \ell_{g^{-1}} \cdot \sqrt{n} < q^{3/4}$. Now, we have $\|h\| \le \sqrt{n} q^{1/2}$, $\|e_i\| \le \sigma' \sqrt{n}$, $|\rho_j| \le \sigma_1^* \sqrt{n}$, $\|b_j^{(1)}\| \le \sigma' \sqrt{n}$, with probability exponentially close to 1, thanks to Lemma 2.3. Now, we have $\|u_i\| = \|e_i + \sum_j \rho_j b_j^{(1)}\| \le \|e_i\| + m_r \max_j |\rho_j| \cdot \|b_j^{(1)}\| \le (m_r+1) \cdot n\sigma_1^* \sigma'$ and $\|c\| = \|\prod_{i=1}^{\kappa} u_i\| \le \sqrt{n}^{\kappa-1} \cdot (\max_i \|u_i\|)^\kappa \le ((m_r+1) n^{1.5} \sigma_1^* \sigma')^\kappa$. Therefore, these two conditions are satisfied if:

$$q > \max\left((n\ell_{g^{-1}})^8, ((m_r+1) \cdot n^{1.5} \sigma_1^* \sigma')^{8\kappa} \right). \tag{16}$$

To satisfy the "negligible probability false positives" zero-testing condition, we need $\|p_{zt}u\|_\infty > q^{3/4}$, for any level-$\kappa$ encoding $u = [c/z^\kappa]_q \in S_\kappa^{(e_L)}$ of $e_L \in R_g$, except with negligible probability $\varepsilon_{zt} = \lambda^{-\omega(1)}$ over the uniform choice of $e_L \in R_g$. By Lemma A.1 with $t = 1$, the facts that $\|c\| < q^{1/8}$, $\|h\| \le \sqrt{n} q^{1/2}$ (see just above), $h \notin \mathcal{I}$ (see just below), and that $\mathcal{I}$ is prime, it follows that, $\|p_{zt}u\|_\infty > q^{3/4}$ for any encoding of a non-zero $e_L \notin I$ (and hence $\varepsilon_{zt} = \Pr[e_L = 0] = 1/|R_g| = O(2^{-n})$), assuming the condition

$$q > (2n\sigma)^4. \tag{17}$$

We have $h \notin \mathcal{I}$, except with probability $O(1/|R/\mathcal{I}|)$ over the choice of h, by Lemma 2.6, when $q = \omega(n\sigma)^2$. Note that thanks to the remark just after Lemma 2.7, we have $|R/I| \ge \sigma_n(\mathrm{rot}(g))^n \ge (\frac{1}{\sqrt{n} \cdot \|g^{-1}\|})^n$. Now, by the $\mathsf{InstGen}$ rejection test, we have $\|g^{-1}\| \le \ell_{g^{-1}}$. Condition (4) finally implies that $|R/I| \ge 2^n$ when $n \ge 8$.

Correctness of extraction. To satisfy the min-entropy extraction condition, we need that the min-entropy of $[p_{zt}u]_q$ is $\ge 2\lambda$. Indeed, any two level-κ encodings $u = [(e_L + gr)/z^\kappa]_q$ and $u' = [(e'_L + gr')/z^\kappa]_q$ of different elements $e_L \ne e'_L \in R_g$ have different extracted elements $\mathrm{MSB}_\ell(p_{zt}u) \ne \mathrm{MSB}_\ell(p_{zt}u')$ as long as $\|[p_{zt}u]_q - [p_{zt}u']_q\|_\infty = \|[p_{zt}(u-u')]_q\|_\infty > 2^{L-\ell+1}$. If that condition is satisfied, then the min-entropy is $\log_2 |R/\mathcal{I}|$. As $|R/I| \ge 2^n$ for $n \ge 8$ (see above), we have $\log_2 |R/\mathcal{I}| \ge n \ge 2\lambda$. We now prove that the condition $\|[p_{zt}(u-u')]_q\|_\infty > 2^{L-\ell+1}$ is satisfied. Since $u - u'$ is an encoding of a non-zero element $e_L - e'_L \in R_g$ this follows, similarly to the zero-testing correctness above, from Lemma A.1 with t satisfying $tq^{3/4} > 2^{L-\ell+1}$. The latter holds with $t = q^{1/4} 2^{-\ell+2}$. The condition $t > 1$ is satisfied by the upper bound (19) on ℓ below, while the condition $q > (2tn\sigma)^4$ is satisfied by the lower bound

$$\ell > \log_2(8n\sigma). \tag{18}$$

To satisfy the "negligible failure probability" extraction condition, we need $\mathrm{MSB}_\ell(p_{zt}u) = \mathrm{MSB}_\ell(p_{zt}u')$ for any two level-κ encodings $u = [(e_L + gr)/z^\kappa]_q$ and $u' = [(e_L + gr')/z^\kappa]_q$ of the same element $e_L \in R_g$, except with negligible probability ε_{ext} over the uniform choice of $e_L \in R_g$. Since $[p_{zt}u]_q = [he_L/g + hr]_q$ and $[p_{zt}u']_q = [he_L/g + hr']_q$ with $\|hr\|_\infty, \|hr'\|_\infty < q^{3/4}$, we can only have $\mathrm{MSB}_\ell(p_{zt}u) \ne \mathrm{MSB}_\ell(p_{zt}u')$ if he_L/g falls within infinity distance $< q^{3/4}$ of a multiple of $2^{L-\ell+1}$, where $L = \lfloor \log_2 q \rfloor$. Under the heuristic assumption that each coefficient of $[he_L/g]_q$ is uniform in $\mathbb{Z}_q$ over the choice of e_L (this heuristic assumption is reasonable from the point of view of entropy; indeed, by the min-entropy condition above, the entropy of $[he_L/g]_q \in R_q$ over the choice of e_L uniformly in $R/\mathcal{I}$, is at least n bit, and this exceeds $\log_2 q$ because of the lattice rule of thumb security requirement $n = \Omega(\lambda \log q)$ in Eq. (15)), we have by a union bound over all n coefficients that this "bad" event occurs with probability $\le \frac{2nq^{3/4}}{2^{L-\ell+1}}$. To make this probability $\le \varepsilon_{ext}$, it suffices to take

$$\ell \le \frac{1}{4} \log_2 q - \log_2(\frac{2n}{\varepsilon_{ext}}). \tag{19}$$

A.2 Review of GGH re-randomization security reduction

To set the background for our result, we review the re-randomization security reduction from the non-canonical problems to their canonical variants, which is implicit in the work of Garg et al. (GGH) [10]. For simplicity, we explain it for the case of Ext-GCDH, although it holds similarly for the other variants GCDH and GDDH.

The first step is to show that re-randomization security goal in Definition 3.4 is satisfied if the strong re-randomization requirement in Definition 3.2 is satisfied. Let $\mathcal{A}$ denote a (T, ε) adversary against problem Ext-GCDH, in which $e_i \hookleftarrow D_{R,\sigma'}$, $u'_i = [e_i \cdot y]_q = [(e_{i,L} + gr'_i)z^{-1}]_q$ with $e_{i,L} = [e_i]_g$, and $u_i = [u'_i + \sum_j \rho_{ij} x_j]_q$ where $\rho_{ij} \hookleftarrow D_{R,\sigma_1^*}$, for $i \in \{0, \ldots, \kappa\}$ and $j \in \{1, \ldots, m_r\}$. Let Game_1 denote this game. Now let Game_3 denote the game in which $e_i \hookleftarrow D_{R,\sigma'}$ and $u_i = [(e_{i,L} + gr_i)z^{-1}]_q$ with $e_{i,L} + gr_i \hookleftarrow D^{(1)}_{\mathrm{can}}(e_{i,L}) = D_{\mathcal{I}+e_{i,L},\sigma_1^*(B^{(1)})^T}$.

Note that the only difference between the two games is the distribution of the randomizers r_i: in Game_1, we have $r_i = r'_i + \sum_j \rho_{ij} r_j^{(1)}$, which has the distribution $D^{(1)}_{u_i}(e_{i,L}, r'_i)$ in Definition 3.2 (over the randomness of ρ_{ij}), while in Game_3, we have r_i sampled from the canonical distribution $(D^{(1)}_{\mathrm{can}}(e_{i,L}) - e_{i,L})/g$. Hence, by the strong re-randomization requirement in Definition 3.2, the statistical distance between the r_i's in the two games is $\leq 2^{-\lambda}$. Therefore, we have that the statistical distance between the distributions of the view of $\mathcal{A}$ in the two games is at most $(\kappa + 1) \cdot 2^{-\lambda}$. Finally, let Game_4 denote the Ext-cGCDH game. The only difference between Game_3 and Game_4 is the distribution of $e_{i,L}$: in Game_3, we have $e_{i,L} = [e_i]_g$ with e_i sampled from $D_{R,\sigma'}$, whereas in Game_4 we have $e_{i,L}$ sampled uniformly from R_g. By Lemma 2.6, if $\sigma' \geq \eta_{\varepsilon_e}(\mathcal{I})$, then the statistical distance between the distributions of $e_{i,L}$ in both games is $\leq 2\varepsilon_e$, so that the statistical distance between the view of $\mathcal{A}$ in both games is $O(\kappa \cdot \varepsilon_e)$. By Lemma 2.4, the latter condition is satisfied if

$$\sigma' = \|g\| \cdot \Omega\left(\sqrt{\log(n\varepsilon_e^{-1})}\right) \geq \sigma\sqrt{n} \cdot \Omega\left(\sqrt{\log(n\varepsilon_e^{-1})}\right). \tag{20}$$

The second step is to show that the strong re-randomization requirement in Definition 3.2 is satisfied, i.e., that the distribution of r_i in Game_3 is statistically close to the distribution of r_i in Game_1. To do so, consider the intermediate game Game_2, in which the distribution of the term $\sum_j \rho_{ij} r_j^{(1)}$ is replaced by $D^{(1)}_{\mathrm{can}}(0)$, so that $r_i = r'_i + w$, where $w \hookleftarrow D^{(1)}_{\mathrm{can}}(0) = D_{\mathcal{I},\sigma_1^*(B^{(1)}))^T}$. There are now two changes to analyze:

- For the change from Game_1 to Game_2, the authors of [10] apply a discrete Gaussian variant of the Leftover Hash Lemma from [2] (see Theorem B.5 in Section 5) to show that $\Delta(\sum_j \rho_{ij} r_j^{(1)} : \rho_{ij} \hookleftarrow D_{\mathbb{Z},\sigma_1^*}; D_{\mathcal{I},\sigma_1^*(B^{(1)})^T}) \leq 2\varepsilon_\rho$ if $m_r = \Omega(n \log n)$ and $\sigma_1^* = \Omega(m_r n^2 \log(1/\varepsilon_\rho))$.
- For the change from Game_2 to Game_3, the authors of [10] argue (informally) that if the randomizer deviation parameter σ_1^* is sufficiently large to "drown" the offset $r'_i \in \mathcal{I}$ by an exponential ratio, i.e., if $\sigma_1^*/\|r'_i\| \geq 2^\lambda$, then the statistical distance between $r'_i + D_{\mathcal{I},\sigma_1^*(B^{(1)})^T}$ and $D_{\mathcal{I}+e_{i,L},\sigma_1^*(B^{(1)})^T}$ is $O(\|r'_i\|/\sigma_1^*) \leq O(2^{-\lambda})$.

Overall, the statistical distance between the views of $\mathcal{A}$ in Game_1 and Game_4 is $\Delta(\mathsf{Game}_1, \mathsf{Game}_4) = O(\kappa \cdot (\varepsilon_\rho + \|r'_i\|/\sigma^* + \varepsilon_e))$. Therefore, algorithm $\mathcal{A}$ solves Ext-cGCDH with run-time $T' = T$ and success probability

$$\varepsilon' \geq \varepsilon - O(\kappa \cdot (\varepsilon_\rho + \|r'_i\|/\sigma^* + \varepsilon_e)), \tag{21}$$

so that the re-randomization security goal of Definition 3.4 is satisfied if

$$\|r_i'\|/\sigma^*, \varepsilon_\rho, \varepsilon_e \;=\; O(\kappa^{-1} \cdot 2^{-\lambda}), \tag{22}$$

and $m_r = \Omega(n \log n)$ and $\sigma' = \|g\| \cdot \Omega\left(\sqrt{\log(n\varepsilon_e^{-1})}\right)$.

Our main contribution is to improve the above analysis, and show how to satisfy the security goal with much better parameters, namely $\|r_i'\|/\sigma^*, \varepsilon_\rho, \varepsilon_e = O(\kappa^{-1})$. In Section 5, we show that we can take $m_r = 2$ in the leftover hash lemma step, between Game_2 and Game_3, using a ring-based variant of the leftover hash lemma from [2]. In Section 4, we develop a better analysis of the drowning step above, between Game_2 and Game_3.

B Review of the discrete leftover hash lemma from [2]

We review the results of [2]. For $X \in \mathbb{Z}^{n\times m}$ and $s > 0$, the authors define the distribution $\mathcal{E}_{X,s} = X \cdot D_{\mathbb{Z}^m,s}$ as the distribution induced by sampling an integer vector $\boldsymbol{v}$ from a discrete spherical Gaussian with parameter s and outputting $\boldsymbol{y} = X\cdot\boldsymbol{v}$. They show that with overwhelming probability over the choice of X, the distribution $\mathcal{E}_{X,s}$ is statistically close to a discrete Gaussian distribution.

Theorem B.1 ([2, Theorem 2]). *For ε negligible in n, let $S \in \mathbb{R}^{n\times n}$ be a matrix such that $s_n = \sigma_n(S) \geq 18K\eta_\varepsilon(\mathbb{Z}^n)$ (for some universal constant $K > 0$), and set $s_1 = \sigma_1(S)$ and $w = s_1/s_n$. Also let m, s be parameters such that $m \geq 10n\log(8(mn)^{1.5}s_1 w)$ and $s' \geq 4mnw\ln(1/\varepsilon)$.*

Then, when choosing the columns of an n-by-m matrix X from the ellipsoid Gaussian over $\mathbb{Z}^n$, $X \hookleftarrow (D_{\mathbb{Z}^n,S})^m$, we have with all but probability $2^{-\Omega(m)}$ over the choice of X, that the statistical distance between $\mathcal{E}_{X,s}$ and the ellipsoid Gaussian $D_{\mathbb{Z}^n,sX^T}$ is bounded by 2ε.

Note that this result has been recently improved in [1], but this improvement is independent from ours. In [1], the authors keep the same distribution $\mathcal{E}_{X,s}$, but obtain weaker conditions under which the result holds. We recall the proof line of [2], as we modify it in our improvement. In [2], the proof of this theorem proceeds by the following three lemmata.

Lemma B.2 ([2, Lemma 9]). *With parameters as above, when drawing the columns of an n-by-m matrix X independently at random from $D_{\mathbb{Z}^n,S}$, we get $X\cdot\mathbb{Z}^n = \mathbb{Z}^n$ with all but probability $2^{-\Omega(m)}$.*

Let $A = A(X) = \{\boldsymbol{v} \in \mathbb{Z}^m : X\cdot\boldsymbol{v} = 0\}$ be the $(m-n)$-dimensional lattice in $\mathbb{Z}^n$ orthogonal to all the rows of X. If the smoothing parameter of A is small, then $\mathcal{E}_{X,s}$ and $D_{\mathbb{Z}^n,sX^T}$ must be close.

Lemma B.3 ([2, Lemma 10]). *Fix X and A as above. If $s \geq \eta_\varepsilon(A)$, then for any point $\boldsymbol{z} \in \mathbb{Z}^n$, the probability mass assigned to $\boldsymbol{z}$ by $\mathcal{E}_{X,s}$ differs from that assigned by $D_{\mathbb{Z}^n,sX^T}$ by at most a factor of $(1-\varepsilon)/(1+\varepsilon)$, namely*

$$\mathcal{E}_{X,s}(\boldsymbol{z}) \in \left[\frac{1-\varepsilon}{1+\varepsilon}, 1\right] \cdot D_{\mathbb{Z}^n,sX^T}(\boldsymbol{z})$$

In particular, if $\varepsilon < 1/3$ then the statistical distance between $\mathcal{E}_{X,s}$ and $D_{\mathbb{Z}^n,sX^T}$ is at most 2ε.

Finally, the authors of [2] show that the smoothing parameter of A is indeed small.

Lemma B.4 ([2, Corollary 3]). *With the parameters above, the smoothing parameter of A satisfies $\eta_\varepsilon(A) \leq 4mnw\ln(1/\varepsilon)$, except with probability $2^{-\Omega(m)}$.*

The following also holds for general lattices.

Theorem B.5 ([2, Theorem 3]). *Let $\Lambda \subseteq \mathbb{R}^n$ be a full-rank lattice and B a matrix whose columns form a basis of Λ. Also let $M \in \mathbb{R}^{n \times n}$ be a full-rank matrix, and denote $S = M(B^T)^{-1}$, $s_1 = \sigma_1(S)$, $s_n = \sigma_n(S)$, and $w = s_1/s_n$. Finally, let ε be negligible in n and m, s be parameters such that $m \geq 10n\log(8(mn)^{1.5}s_1 w)$ and $s \geq 4mnw\ln(1/\varepsilon)$. If $s \geq \eta_\varepsilon(\mathbb{Z}^n)$, then when choosing the columns of an n-by-m matrix X from the ellipsoid Gaussian over Λ, $X \hookleftarrow (D_{\Lambda,M})^m$, we have with all but probability $2^{-\Omega(m)}$ over the choice of X, that the statistical distance between $\mathcal{E}_{X,s}$ and the ellipsoid Gaussian D_{Λ,sX^T} is bounded by 2ε.*