# Structural Lattice Reduction: Generalized Worst-Case to Average-Case Reductions and Homomorphic Cryptosystems

Nicolas Gama[1,2], Malika Izabachène[3], Phong Q. Nguyen[4,5], and Xiang Xie[6]

[1] Laboratoire de Mathématiques de Versailles, UVSQ, CNRS, Université Paris-Saclay
78035 Versailles, France
[2] Inpher, Lausanne, Switzerland
[3] CEA, LIST, 91191 Gif-sur-Yvette Cedex, France
[4] Inria, France
[5] CNRS/JFLI and the University of Tokyo, Japan
[6] Huawei Technologies, China

**Abstract.** In lattice cryptography, worst-case to average-case reductions rely on two problems: Ajtai's SIS and Regev's LWE, which both refer to a very small class of random lattices related to the group $G = \mathbb{Z}_q^n$. We generalize worst-case to average-case reductions to all integer lattices of sufficiently large determinant, by allowing $G$ to be any (sufficiently large) finite abelian group. Our main tool is a novel generalization of lattice reduction, which we call structural lattice reduction: given a finite abelian group $G$ and a lattice $L$, it finds a short basis of some lattice $\bar{L}$ such that $L \subseteq \bar{L}$ and $\bar{L}/L \simeq G$. Our group generalizations of SIS and LWE allow us to abstract lattice cryptography, yet preserve worst-case assumptions: as an illustration, we provide a somewhat conceptually simpler generalization of the Alperin-Sheriff-Peikert variant of the Gentry-Sahai-Waters homomorphic scheme. We introduce homomorphic mux gates, which allows us to homomorphically evaluate any boolean function with a noise overhead proportional to the square root of its number of variables, and bootstrap the full scheme using only a linear noise overhead.

## 1 Introduction

A lattice is a discrete subgroup of $\mathbb{R}^m$. Nearly two decades after its introduction, lattice-based cryptography has emerged as a credible alternative to classical public-key cryptography based on factoring or discrete logarithm. It offers new properties (such as security based on worst-case assumptions) and new functionalities, such as noisy multilinear maps and fully-homomorphic encryption. The worst-case guarantees of lattice-based cryptography come from two problems: Ajtai's *short integer solution* (SIS) [1] and Regev's *learning with errors* (LWE) [37]. These average-case problems are provably as hard as solving certain lattice problems in the worst case, such as GapSVP (the decision version of the shortest vector problem) and SIVP (finding short lattice vectors).

As noted by Micciancio [25], the SIS problem can be defined as finding short vectors in a random lattice from a class $\mathcal{A}_{n,m,q}$ of $m$-dimensional integer lattices related to the finite abelian group $G = \mathbb{Z}_q^n$, where $n$ is the dimension of the worst-case lattice problem and $q$ needs to be sufficiently large: any $\mathbf{g} = (g_1, \ldots, g_m) \in G^m$ chosen uniformly at random defines a lattice $\mathcal{L}_{\mathbf{g}} \in \mathcal{A}_{n,m,q}$ formed by all $\mathbf{x} = (x_1, \ldots, x_m) \in \mathbb{Z}^m$ s.t. $\sum_{i=1}^m x_i g_i = 0$ in $G$; and SIS asks, given $\mathbf{g}$, to find a short (nonzero) $\mathbf{x} \in \mathcal{L}_{\mathbf{g}}$. The class $\mathcal{A}_{n,m,q}$ has an algebraic meaning: for suitable parameters, the distribution of $\mathcal{L}_{\mathbf{g}}$ is statistically close to the uniform distribution over the finite set $\mathcal{L}_{G,m}$ of all full-rank lattices $L \subseteq \mathbb{Z}^m$ such that $\mathbb{Z}^m/L \simeq G$. This suggests that Ajtai's lattices are very rare among all integer lattices: in fact, Nguyen and Shparlinski [31] recently showed that the set $\cup_{G \text{ cyclic}} \mathcal{L}_{G,m}$ of all full-rank integer lattices $L \subseteq \mathbb{Z}^m$ such that $\mathbb{Z}^m/L$ is cyclic (unlike $\mathbb{Z}_q^n$) has natural density $1/[\zeta(6) \prod_{k=4}^m \zeta(k)] \approx 85\%$ (for large $m$), which implies that Ajtai's classes $\mathcal{A}_{n,m,q}$ form a minority among all integer lattices.

This motivates the natural question of whether other classes of random lattices enjoy similar worst-case to average-case reductions: if we call GSIS the SIS generalization (introduced by Micciancio [25, Def 5.2]) to any finite abelian group $G$, does GSIS have similar properties as SIS for other groups than $G = \mathbb{Z}_q^n$? This would imply that the random lattices of $\mathcal{L}_{G,m}$ are also hard. Ajtai (in the proceedings version of [1]) and later Regev [36] noticed that the choice $G = \prod_{i=1}^n \mathbb{Z}_{q_i}$ where the $q_i$'s are distinct prime numbers of similar bit-length also worked. Micciancio [25] gave another choice of $G$, to obtain a better worst-case to average-case connection (at that time): his $G$ is actually constructed by an algorithm [25, Lemma 2.11] given as input a very special lattice (for which solving the closest vector problem is easy); if the input lattice is $\mathbb{Z}^n$, then $G = (\mathbb{Z}_q)^n$. However, all these choices of $G$ are very special, and it was unknown if the hardness properties held outside a small family of finite abelian groups.

A similar question can be asked for LWE, which is known as a dual problem of SIS, and has been used extensively in lattice-based encryption. However, in order to define GLWE by analogy with GSIS, we need to change the usual definition of LWE based on linear algebra. Any finite abelian group $G$ is isomorphic to its dual group $\hat{G}$ formed by its characters, *i.e.* homomorphisms from $G$ to the torus $\mathbb{T} = \mathbb{R}/\mathbb{Z}$. We define search-GLWE as the problem of learning a character $\hat{s} \in \hat{G}$ chosen uniformly at random, given noisy evaluations of $\hat{s}$ at (public) random points $g_1, \ldots, g_m \in G$, namely one is given $g_i$ and a "Gaussian" perturbation of $\hat{s}(g_i)$ for all $1 \leq i \leq m$. Decisional-GLWE is defined as the problem of distinguishing the previous "Gaussian" perturbations of $\hat{s}(g_i)$ from random elements in $\mathbb{T}$. If $G = (\mathbb{Z}_q)^n$, it can be checked that GLWE is LWE. If $G = \mathbb{Z}_p$ for some large prime $p$, search-GLWE is a randomized version of Boneh-Venkatesan's *Hidden Number Problem* (HNP) [8] (introduced to study the bit-security of Diffie-Hellman key exchange, but also used in side-channel attacks on discrete-log based signatures [30]), which asks to recover a secret number $s \in \mathbb{Z}_p$, given random $t_1, \ldots, t_m$ chosen uniformly from $\mathbb{Z}_p$ and approximations of each $st_i \bmod p$. Here, randomized means that the approximations given are "Gaussian" perturbations of $st_i \bmod p$. Thus, GLWE captures LWE

and the HNP as a single problem, instantiated with different groups. Alternatively, GLWE can be viewed as a lattice problem: solving a randomized version of bounded distance decoding (with "Gaussian" errors) for the dual lattice of $\mathcal{L}_{\mathbf{g}}$.

Our results. We show that the worst-case to average-case reductions for SIS and LWE (search and decisional) can be generalized to GSIS and GLWE, provided that $G$ is any sufficiently large finite abelian group, *e.g.* of order $n^{\Omega(\max(n,\mathrm{rank}(G)))}$ if $n$ is the dimension of the worst-case lattice problem and $\mathrm{rank}\,(G)$ denotes the minimal size of a generating set for $G$: note that the order of $G$ is the determinant of the average-case lattice. For GSIS and search-GLWE, our reductions are direct from worst-case problems. We transfer decisional-LWE hardness results to decisional-GLWE by generalizing the modulus-dimension switching technique of Brakerski *et al.* [11].

We believe that our results offer a cleaner high-level picture of worst-case to average-case reductions: previous work tend to focus on quantitative aspects (such as decreasing the worst-case approximation factor, or the parameter $q$, *etc.*), including work on the ring setting, where one introduces a trade-off between security and efficiency. The ring setting offers more efficient primitives but requires (much) stronger worst-case assumptions: in the ring variants of SIS and LWE, the worst-case lattices are restricted to classes of very special lattices known as ideal lattices.

Our reductions are based on a new tool, which we call structural lattice reduction, and which is of independent interest: Becker *et al.* [5] recently used it to design new exponential-space algorithms for lattice problems. In lattice reduction, one is given a full-rank lattice $L \subseteq \mathbb{Z}^n$ and wants to find a short basis of $L$. In our structural lattice reduction, one is further given a finite abelian group $G$ of rank $\leq n$, and wants to find a short basis of some overlattice $\bar{L}$ of $L$ such that $\bar{L}/L \simeq G$ effectively, *i.e.* there exists an efficiently computable surjective map $\varphi$ from $\bar{L}$ to $G$ with $\ker \varphi = L$. Our key point is that previous worst-case to average-case reductions (*e.g.* [20,11]) implicitly used a trivial case[7] of structural lattice reduction: if $B$ is a short basis of a full-rank lattice $L \subseteq \mathbb{Z}^n$ and $q$ is an integer, then $q^{-1}B$ is a short basis of the lattice $\bar{L} = q^{-1}L$ such that $\bar{L}/L \simeq \mathbb{Z}_q^n$, which summarizes the importance of $\mathbb{Z}_q^n$ in SIS and LWE.

Our GSIS reduction shows that in some sense all integer lattices are hard. Indeed, the set of full-rank lattices $L \subseteq \mathbb{Z}^m$ (of sufficiently large co-volume $\geq n^{\Omega(m)}$) can be partitioned based on the finite abelian group $\mathbb{Z}^m/L$, and the reduction implies that each partition cell $\mathcal{L}_{G,m}$ has this worst-case to average-case property: finding short vectors in a lattice chosen uniformly at random from $\mathcal{L}_{G,m}$ is as hard as finding short vectors in any integer lattice of dimension $n$.

Consider the special case $G = \mathbb{Z}_p$ for a large prime $p$. Then our GSIS reduction provides the first hardness results for the random lattices in $\mathcal{L}_{\mathbb{Z}_p,m}$ used in many experiments [18,14] to benchmark lattice reduction algorithms, as well as in Darmstadt's SVP internet challenges. And our GLWE reduction provides

[7] There is a more technical reduction implicitly proposed in [25], but unfortunately too restrictive on the choice of $G$

a general hardness result for the HNP: previously, [11, Cor 3.4] established the hardness for HNP when the large prime $p$ is replaced by $q^n$ where $q$ is smooth.

Finally, our generalizations of SIS and LWE allow us to abstract (the many) lattice-based schemes based on SIS and/or LWE, where the role of $G = (\mathbb{Z}_q)^n$ was not very explicit in most descriptions (typically based on linear algebra). We believe such an abstraction can have several benefits. First, it can clarify analyses and designs: the El Gamal cryptosystem is arguably better described with an arbitrary group $G$, rather than by focusing on the historical choice $G = \mathbb{Z}_p^*$; comparisons and analogies with "traditional" public-key cryptography based on factoring or discrete logarithm will be easier. We illustrate this point by providing a somewhat conceptually simpler GLWE-based generalization of the Alperin-Sheriff-Peikert variant [2] of the Gentry-Sahai-Waters homomorphic scheme [21]: this generalization becomes essentially as simple as trapdoor-based fully-homormophic encryption proposals such as [38]. It is based on a GLWE variant of El Gamal encryption, which naturally generalizes Regev's LWE encryption [37]. We also provide a new decryption circuit based on Mux gates, which can bootstrap the system with a polynomial noise overhead, and is arguably simpler than [2]. Second, it opens up the possibility of obtaining more efficient schemes using different choices of $G$ than $G = (\mathbb{Z}_q)^n$. We do not claim that there are better choices than $(\mathbb{Z}_q)^n$, but such a topic is worth investigating, which we leave to future work. Many factors influence efficiency: trapdoor generation, hashing, efficiency of the security reduction, *etc.* For instance, hashing onto $\mathbb{Z}_p$ can sometimes be more efficient than onto $(\mathbb{Z}_q)^n$ for large $n$, which could be useful in certain settings, like digital signatures.

Furthermore, our abstraction may also be helpful to better understand attacks on GSIS and GLWE. For instance, there are similarities between Bleichenbacher's algorithm [6] for HNP and the BKW algorithm [7] for LWE: by viewing LWE and HNP as two different instances of the same problem GLWE, one can focus on the main ideas. And we note that among several classes of random lattices having a worst-case to average-case reduction, it could be that some are weaker than others, when it comes to the best attack known.

Related work. Baumslag *et al.* also introduced in [4] group generalizations of LWE for non-commutative groups, but did not obtain hardness result. [16] showed a self-reducibility property for some special non-commutative groups.

Open problems. Similarly to [11], our strongest hardness result for decisional-GLWE bypasses search-GLWE: a direct search-to-decision equivalence for all sufficiently large $G$ is open. Adapting structural lattice reduction to the ring setting is open: current ring results only address the average-case hardness of very few classes of lattices, and it would be interesting to tackle more classes. Our reductions require the order of $G$ to be large compared to the worst-case lattice dimension, and we would like to minimize this constraint: the GLWE case $G = \mathbb{Z}_2^n$ is essentially LPN, whose hardness is open; here, the order $2^n$ does not grow quickly enough with respect to the rank $n$ for our reduction. On the other hand, Micciancio and Peikert [27] recently decreased $q$ for SIS.

ROADMAP. Sect. 2 gives background. Sect. 3 presents our group generalizations of SIS and LWE. Sect. 4 presents structural lattice reduction. Sect. 5 and 6 show hardness of GSIS and decisional-GLWE. In Sect. 7, we give an example of abstracting lattice cryptography: El Gamal-like encryption and fully-homomorphic encryption from GLWE. Detailed missing proofs can be found in the full version of the paper [17]. In particular, we compare structural reduction with previous work of Ajtai [1] and Micciancio [25]: and show that all previous SIS reductions can be captured by our overlattice framework.

## 2 Background and Notation

$\mathbb{Z}_q$ denotes $\mathbb{Z}/q\mathbb{Z}$. We use row notation for vectors and matrices. $I_n$ is the $n \times n$ id. matrix. A function $\mathrm{negl}(n)$ is *negligible* if it vanishes faster than any inverse polynomial. $\|B\| = \max_{1 \le i \le n} \|\mathbf{b}_i\|$ is the maximal row norm of a matrix $B$.

*Lattices.* A *lattice* $L$ is of the form $L(B) = \{\sum_{i=1}^{n} \alpha_i \mathbf{b}_i,\ \alpha_i \in \mathbb{Z}\}$ for some *basis* $B = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ of linearly independent vectors in $\mathbb{R}^m$. If $L \subseteq \mathbb{Z}^m$, $L$ is an *integer lattice*. The dimension $n$ of $\mathsf{span}(L)$ is the *dimension* $\mathsf{dim}(L)$ of $L$. The *(co)-volume* $\mathsf{vol}(L)$ is $\sqrt{\det(BB^t)}$ for any basis $B$ of $L$. For $1 \le i \le \mathsf{dim}(L)$, $\lambda_i(L)$ is the $i$-th minimum of $L$, (smallest radius of the 0-ball containing at least $i$ linearly indep. lattice vectors) The *dual lattice* $L^\times$ is the set of all $\mathbf{u} \in \mathsf{span}(L)$ s.t. $\langle \mathbf{u}, \mathbf{v} \rangle \in \mathbb{Z}$ for all $\mathbf{v} \in L$. If $B$ is a basis of $L$, its *dual basis* $B^\times = (BB^t)^{-1}B$ is a basis of $L^\times$. For a factor $\gamma = \gamma(n) \ge 1$, $\mathrm{GapSVP}_\gamma$ asks, given $d \ge 0$ and a basis $B$ of an $n$-dim lattice $L$, to decide if $\lambda_1(L) \le d$ or $\lambda_1(L) > \gamma d$. $\mathrm{ApproxSIVP}_\gamma$ asks a full-rank family of lattice vectors of norm $\le \gamma \lambda_n(L)$.

*Gram-Schmidt Orthogonalization (GSO).* The GSO of a lattice basis $B = (\mathbf{b}_1, ..., \mathbf{b}_n)$ is the unique decomposition $B = \mu \cdot D \cdot Q$, where $\mu$ is a lower triangular matrix with unit diagonal, $D$ is a positive diagonal matrix, and $Q$ has orthonormal rows. We let $B^* = DQ$ whose $i$-th row $\mathbf{b}_i^*$ is $\pi_i(\mathbf{b}_i)$, where $\pi_i$ denotes the orthogonal projection of $\mathbf{b}_i$ over $\mathsf{span}\{\mathbf{b}_1, \ldots, \mathbf{b}_{i-1}\}^\perp$. We use the notation $B_{[i,j]}$ for the block $[\pi_i(\mathbf{b}_i), \ldots, \pi_i(\mathbf{b}_j)]$. If $B^\times$ is the dual basis of $B$ and $(B^\times)^*$ denotes its GSO matrix, then $\|(\mathbf{b}_i^\times)^*\| \cdot \|\mathbf{b}_{n-i+1}^*\| = 1$ for $1 \le i \le n$.

*(Explicit) Finite abelian groups.* Any finite abelian group $G$ is isomorphic to a product $\prod_{i=1}^{k} \mathbb{Z}_{q_i}$ of cyclic groups. We call *rank* of $G$ the minimal number of cyclic groups in such decompositions: this should not be confused with the rank of an abelian group. We say that $G$ is *explicit* if one knows $q_1, \ldots, q_k \in \mathbb{N}$ and an isomorphism $\prod_{i=1}^{k} \mathbb{Z}_{q_i} \to G$ computable in poly-time: wlog $k$ is the rank and $q_{i+1} | q_i$. The isomorphism induces $k$ generators $e_1, \ldots, e_k \in G$ s.t. $G = \langle e_1 \rangle \oplus \cdots \oplus \langle e_k \rangle$ and each $e_i$ has order $q_i$. If the inverse of the isomorphism is also computable in polynomial time, we say that $G$ is *fully-explicit*.

*Overlattices.* When a lattice $\bar{L}$ contains a sublattice $L$ of the same dimension $n$, $\bar{L}$ is an *overlattice* of $L$. Then $\bar{L}/L$ is a finite abelian group of rank $\le n$ and

order $\mathrm{vol}(L)/\mathrm{vol}(\bar{L})$. Then we note $\bar{L}/L \overset{\varphi}{\simeq} G$ for some $\varphi$, *i.e.* $\varphi : \bar{L} \to G$ is a surjective morphism s.t $\ker \varphi = L$.

*Lattice reduction.* Cai [13] introduced the *basis length* of a lattice $L$ as $\mathrm{bl}(L) = \min_{\text{basis } B} \|B^*\|$. Then: $\lambda_n(L) \geq \mathrm{bl}(L) \geq \lambda_n(L)/\sqrt{n}$, $\mathrm{bl}(L) \geq \lambda_1(L)$, and $\mathrm{bl}(L) \geq \mathrm{vol}(L)^{1/n}$. Lattice reduction can find bases $B$ with small $\|B^*\|$. A basis B is LLL-reduced [23] with factor $\varepsilon_{\text{LLL}} \geq 0$ if its GSO satisfies $|\mu_{i,j}| \leq \frac{1}{2}$ for all $1 \leq j < i$ and $\|\mathbf{b}_i^*\|^2 \leq (1+\varepsilon_{\text{LLL}})(\left\|\mathbf{b}_{i+1}^*\right\|^2 + \mu_{i+1,i}\|\mathbf{b}_i^*\|^2)$. Then it is folklore that: $\|B^*\| \leq \left((1+\varepsilon_{\text{LLL}})\sqrt{4/3}\right)^{(n-1)/2} \mathrm{bl}(L)$. Given $\varepsilon_{\text{LLL}} > 0$ and a basis $B$ of a lattice $L \subseteq \mathbb{Z}^n$, LLL [23] outputs an LLL-reduced basis of factor $\varepsilon_{\text{LLL}}$ in time polynomial in $1/\varepsilon_{\text{LLL}}$ and size$(B)$. Usually, $(1+\varepsilon_{\text{LLL}})\sqrt{4/3} = \sqrt{2}$ or $\varepsilon_{\text{LLL}} = 1/\mathrm{poly}(n)$.

### 2.1 Gaussian Measures

The statistical distance between two distributions $\mathcal{P}$ and $\mathcal{Q}$ over a domain $X$ is $\Delta(\mathcal{P},\mathcal{Q}) = \frac{1}{2}\int_{a\in X} |\mathcal{P}(a) - \mathcal{Q}(a)|da$ or $\frac{1}{2}\sum_{a\in X} |\mathcal{P}(a) - \mathcal{Q}(a)|$ when $X$ is discrete. $\mathcal{P}$ and $\mathcal{Q}$ are (statistically) $\varepsilon$-indistinguishable if $\Delta(\mathcal{P},\mathcal{Q}) < \varepsilon$. We write $\mathbf{y} \leftarrow \mathcal{P}$ (resp. $\leftarrow_\varepsilon \mathcal{P}$) for a sample $\mathbf{y}$ from the distrib. $\mathcal{P}$ (resp. a distribution $\varepsilon$-indistinguishable from $\mathcal{P}$). And $\leftarrow_{\approx}$ means $\leftarrow_\varepsilon$ for some negligible function $\varepsilon$.

*Gaussian Distributions.* The *Gaussian Distribution* (over $\mathbb{R}^n$) $\mathcal{D}_{\mathbb{R}^n,\sigma,\mathbf{c}}$ centered at $\mathbf{c} \in \mathbb{R}^n$ of parameter $\sigma \in \mathbb{R}_{\geq 0}$ has a density function proportional to $\rho_{\mathbb{R}^n,\sigma,\mathbf{c}}(\mathbf{x}) = \exp\left(-\pi\|\mathbf{x}-\mathbf{c}\|^2/\sigma^2\right)$. If $\mathbf{c}$ is omitted, then $\mathbf{c} = 0$. For any countable subset $C \subseteq \mathbb{R}^n$ (a lattice $L$ or a coset $\mathbf{x}+L$), $\rho_{\mathbb{R}^n,\sigma,\mathbf{c}}(C)$ is $\sum_{\mathbf{u}\in C}\rho_{\mathbb{R}^n,\sigma,\mathbf{c}}(\mathbf{u})$. The *discrete Gaussian distribution* $\mathcal{D}_{C,\sigma,\mathbf{c}}$ over a lattice or coset $C \subset \mathbb{R}^n$ is $\mathcal{D}_{C,\sigma,\mathbf{c}}(\mathbf{x}) = \rho_{\mathbb{R}^n,\sigma,\mathbf{c}}(\mathbf{x})/\rho_{\mathbb{R}^n,\sigma,\mathbf{c}}(C)$ where $\mathbf{x} \in C$. One can sample efficiently the discrete Gaussian distribution within negligible distance [20,35] or exactly [11]:

**Lemma 1.** *There is a poly-time algorithm which, given $\mathbf{c} \in \mathbb{Q}^n$, a basis $B$ of a lattice $L \subseteq \mathbb{Q}^n$ and $\sigma \geq \|B^*\| \cdot \sqrt{\ln(2n+4)/\pi}$, samples the dist. $\mathcal{D}_{L,\sigma,\mathbf{c}}$.*

Reciprocally, a short lattice basis is derived from short discrete Gaussian samples:

**Proposition 1** *(Cor. of [36, Lemma 14]) Let $\varepsilon > 0$ and $L(B)$ be an $n$-dim lattice. Given $m = O(n)$ indep. samples $\mathbf{y}_i \leftarrow_\varepsilon \mathcal{D}_{L,s_i}$ s.t. $\sqrt{2}\eta_\varepsilon(L) \leq s_i \leq \sigma$, $1 \leq i \leq m$, one can compute in poly-time a basis $C$ of $L$ s.t. $\|C^*\| \leq \sqrt{n/2\pi}\cdot\max_i s_i$.*

*Modular Distributions and Smoothing Parameter.* The distributions $\mathcal{D}_{\mathbb{R}^n,\sigma,c}$ and $\mathcal{D}_{\bar{L},\sigma,c}$ over an overlattice $\bar{L} \supseteq L$ can be projected modulo $L$: $\mathcal{D}_{\mathbb{R}^n/L,\sigma,c}$ (resp. $\mathcal{D}_{\bar{L}/L,\sigma,c}$) has density $\mathcal{D}_{\mathbb{R}^n,\sigma,\mathbf{c}}(\mathbf{x}+L)$ for $\mathbf{x} \in \mathbb{R}^n/L$ (resp. $\bar{L}/L$). Both $\mathcal{D}_{\mathbb{R}^n/L,\sigma}$ and $\mathcal{D}_{\bar{L}/L,\sigma}$ converge (uniformly) to the uniform distribution when $\sigma$ increases. This is quantified by the *smoothing parameter* $\eta_\varepsilon(L)$ [28], *i.e.* the minimal $\sigma > 0$ for $\varepsilon > 0$ s.t. $\rho_{\mathbb{R}^n,\frac{1}{\sigma}}(L^\times \setminus \{0\}) \leq \varepsilon$, *i.e.* $\left\|\mathcal{D}_{\mathbb{R}^n/L,\sigma}(\mathbf{x}+L) - \frac{1}{\mathrm{vol}(L)}\right\|_\infty \leq \frac{\varepsilon}{\mathrm{vol}(L)}$:

**Lemma 2 (see Cor 2.8 of [20]).** *If $\bar{L}$ is an overlattice of $L$, $\varepsilon \in (0,1/2)$, $\sigma \geq \eta_\varepsilon(L)$ and $\mathbf{c} \in \mathbb{R}^n$, then $\mathcal{D}_{\bar{L}/L,\sigma,\mathbf{c}+L}$ is within stat. distance $\leq 2\varepsilon$ from the uniform distribution over $\bar{L}/L$.*

For any $n$-dim basis $B$, $\eta_\varepsilon(L(B)) \le \eta_\varepsilon(L(B^*)) \le \eta_\varepsilon(\mathbb{Z}^n) \cdot \|B^*\|$ where $\eta_\varepsilon(\mathbb{Z}^n) \le \sqrt{\log\left(2n \cdot (1+\frac{1}{\varepsilon})\right)/\pi}$. In particular, $\eta_\varepsilon(L) \le \eta_\varepsilon(\mathbb{Z}^n) \cdot \mathrm{bl}(L)$. Finally, we give a technical lemma (proved in App. A.2 of the full version [17]), analogous to [37,35].

**Lemma 3.** *Let $\mathbb{K} = \mathbb{R}$ or $\mathbb{T}$. Let $c \in \mathbb{R}$, $\mathbf{u} \in \mathbb{R}^n$, $\alpha, \sigma \in \mathbb{R}_{\ge 0}$, $\varepsilon \in (0, 1/2)$ and $\mathbf{z} + L$ be a coset of an $n$-dim lattice $L \subseteq \mathbb{R}^n$. Assume that $\left(\frac{1}{\sigma^2} + \frac{\|\mathbf{u}\|^2}{\alpha^2}\right)^{-1/2} \ge \eta_\varepsilon(L)$. Then $\mathcal{D}_{\mathbb{K},\alpha,c+\langle \mathbf{u},\mathbf{v}\rangle}$ where $\mathbf{v} \leftarrow \mathcal{D}_{\mathbf{z}+L,\sigma}$ is within statistical distance $\le 4\varepsilon$ from $\mathcal{D}_{\mathbb{K},\sqrt{\alpha^2+\sigma^2\|\mathbf{u}\|^2},c}$. This still holds when $\mathbb{K} = \frac{1}{N}\mathbb{Z}$ or $\frac{1}{N}\mathbb{Z}/\mathbb{Z}$ if $\alpha \ge \eta_\varepsilon(\frac{1}{N}\mathbb{Z})$.*

## 3 Lattice Factor Groups and Generalizations of SIS/LWE

### 3.1 Lattice Factor Groups

If $L$ is a full-rank lattice $\subseteq \mathbb{Z}^m$, its factor group $\mathbb{Z}^m/L$ is a finite abelian group of order $\mathrm{vol}(L)$. For any finite abelian group $G$, denote by $\mathcal{L}_{G,m}$ the (finite) set of full-rank lattices $L \subseteq \mathbb{Z}^m$ such that $\mathbb{Z}^m/L \simeq G$. The following elementary characterization of $\mathcal{L}_{G,m}$ is a consequence of [33]:

**Theorem 1.** *Let $G$ be a finite abelian group and $L$ be a full-rank lattice in $\mathbb{Z}^m$. Then $L \in \mathcal{L}_{G,m}$ if and only if $G$ has rank $\le m$ and there exists $\mathbf{g} = (g_1, \ldots, g_m) \in G^m$ s.t. the $g_i$'s generate $G$ and $L = \mathcal{L}_{\mathbf{g}}$ where $\mathcal{L}_{\mathbf{g}} = \{(x_1, \ldots, x_m) \in \mathbb{Z}^m \text{ s.t. } \sum_{i=1}^m x_i g_i = 0 \text{ in } G\}$.*

Given $G$, Alg. 5 (App. A) samples efficiently lattices from the uniform distribution over $\mathcal{L}_{G,m}$, and its correctness follows from Lemma 4. Previously, efficient sampling was only known for $G = \mathbb{Z}_p$ for large prime $p$ [22].

**Lemma 4.** *Let $G$ be a finite abelian group. Let $\mathbf{g} = (g_1, \ldots, g_m) \in G^m$ be such that the $g_i$'s generate $G$. Let $\mathbf{h} = (h_1, \ldots, h_m) \in G^m$. Then $\mathcal{L}_{\mathbf{g}} = \mathcal{L}_{\mathbf{h}}$ if and only if there is an automorphism $\psi$ of $G$ such that $h_i = \psi(g_i)$ for all $1 \le i \le m$. In such a case, $\psi$ is uniquely determined.*

We note that several implementations of lattice-based cryptography (such as [19]) implicitly used lattices in $\mathcal{L}_{G,m}$ for some large cyclic group $G$. Recently, Nguyen and Shparlinski [31] showed that such lattices are dominant: the set $\cup_{G \text{ cyclic}} \mathcal{L}_{G,m}$ of all full-rank integer lattices $L \subseteq \mathbb{Z}^m$ such that $\mathbb{Z}^m/L$ is cyclic has natural density $1/[\zeta(6)\prod_{k=4}^m \zeta(k)] \approx 85\%$ (for large $m$).

### 3.2 The Group-SIS Problem (GSIS)

Micciancio [25] introduced the Homogeneous SIS problem which is a natural generalization of SIS to an arbitrary finite abelian group $G$. In this paper, we call it *Group-SIS* problem (GSIS). The parameters are $m \ge 1$ and a bound $\beta > 0$. One picks $\mathbf{g} = (g_1, \ldots, g_m) \in G^m$ uniformly at random. $\mathrm{GSIS}(G, m, \beta)$ asks to find a non-zero vector $\mathbf{x} \in \mathbb{Z}^m$ s.t. $\sum_{i=1}^m x_i g_i = 0$ and $\|\mathbf{x}\| \le \beta$. In other

words, GSIS asks to find short vectors in random relation lattices $\mathcal{L}_{\mathbf{g}} = \{\mathbf{x} \in \mathbb{Z}^m \text{s.t.} \sum_{i=1}^m x_i g_i = 0\}$. For instance, GSIS$(\mathbb{Z}_q^n, m, \beta)$ is SIS, and GSIS$(\mathbb{Z}_q, m, \beta)$ is finding short vectors in random $m$-dimensional co-cyclic lattices of volume $q$. If $\#G$ denotes the order of $G$, the existence of a GSIS-solution is guaranteed if $\beta \geq \sqrt{m}(\#G)^{1/m}$.

GSIS is connected to $\mathcal{L}_{G,m}$ as follows. As soon as $m \geq n + 2\log\log\#G + 5$ (resp. $m > 2\log\#G + 2$), $g_1, \ldots, g_m$ generate $G$ with probability $\geq 1/e$ [24,32] (resp. $\geq 1 - 1/\#G$), in which case $\mathbb{Z}^m/\mathcal{L}_{\mathbf{g}} \simeq G$. In particular, if $m > 2\log\#G + 2$, the distribution of GSIS lattices $\mathcal{L}_{\mathbf{g}}$ is statistically close to the distribution of Alg. 5, and therefore the uniform distribution over $\mathcal{L}_{G,m}$, in which case GSIS is equivalent to finding short vectors in random lattices from $\mathcal{L}_{G,m}$.

Finally, we note that to establish hardness of GSIS, it suffices to focus on low-rank groups $G$. Indeed, if $G' = G \times H$ for some groups $G, H$, then GSIS over $G$ can trivially be reduced to GSIS over $G'$, by "projecting" $G'$ to $G$.

### 3.3 The Group-LWE Problem (GLWE)

We introduce the *Group-LWE* problem (GLWE), using the torus $\mathbb{T} = \mathbb{R}/\mathbb{Z}$ and a finite abelian group $G$. Let $\hat{G}$ be the dual group of homomorphisms $G \to \mathbb{T}$: it is isomorphic to $G$ but not canonically. If $G$ is explicit, $G = \oplus_{i=1}^k \langle e_i \rangle$ where $e_i$ has order $q_i$, and $\hat{G}$ is generated by $\hat{e}_1, \ldots, \hat{e}_k$ defined as $\hat{e}_i(\sum_{j=1}^k \alpha_j e_j) = \alpha_i/q_i \mod 1$ where $0 \leq \alpha_j < q_j$.

Let $\mathcal{S}$ be a known distribution over $\hat{G}$. Search-GLWE is the problem of learning a character $\hat{s} \in \hat{G}$ picked from $\mathcal{S}$, given noisy evaluations of $\hat{s}$ at (public) random points $a_1, \ldots, a_m \in G$, namely one is given (for all $i$'s) $a_i$ and a "Gaussian" perturbation of $\hat{s}(a_i)$. Like LWE, several noise distributions are possible. As in [37], we focus on the continuous distribution where $\hat{s}(a)$ is shifted by an error $e \leftarrow \mathcal{D}_{\mathbb{R},\alpha}$. These distributions need to be discretized in order to have a finite representation. In App. B.4 of the full version, we present discrete versions and show that they are at least as hard as the continuous version for suitable parameters, which explains why we only consider the continuous GLWE problem in the rest:

**Definition 1.** *$G = \oplus_{i=1}^k \langle e_i \rangle$ is an expl. finite abelian group, $\alpha > 0$ and $\hat{s} \in \hat{G}$.*

- *$A_{G,\alpha}(\hat{s})$ is the distribution over $G \times \mathbb{T}$ defined by choosing $a \in G$ uniformly at random, setting $b \leftarrow \mathcal{D}_{\mathbb{T},\alpha,\hat{s}(a)}$, and outputting $(a, b) \in G \times \mathbb{T}$.*
- *Search-*GLWE$_{G,\alpha}(\mathcal{S})$ *asks to find $\hat{s}$ from $A_{G,\alpha}(\hat{s})$ for a fixed $\hat{s} \leftarrow \mathcal{S}$ given arbitrarily many independent samples. By finding $\hat{s}$, we mean finding $s_i \in \mathbb{Z}$ s.t. $\hat{s} = \sum_{i=1}^k s_i \hat{e}_i$.*
- *Decisional-*GLWE$_{G,\alpha}(\mathcal{S})$ *asks to distinguish $A_{G,\alpha}(\hat{s})$ from the uniform distribution over $G \times \mathbb{T}$ for a fixed $\hat{s}$ sampled from $\mathcal{S}$ given arbitrarily many independent samples.*
- *For $0 < \alpha < 1$, (Search) Decisional-*GLWE$_{G,\leq\alpha}(\mathcal{S})$ *is the problem of solving (Search) Decisional-*GLWE$_{G,\beta}(\mathcal{S})$ *for any $\beta \leq \alpha$ respectively,* i.e. *when the noise parameter is unknown yet $\leq \alpha$, by analogy with LWE.*

*Search*-GLWE$_{G,m,\alpha}(\mathcal{S})$ *and Decisional*-GLWE$_{G,m,\alpha}(\mathcal{S})$ *denote the variants where the algorithms have a bounded number $m$ of samples. If $\mathcal{S}$ is omitted, it is the uniform distribution over $\hat{G}$.*

If $G = \mathbb{Z}_q^n$, the canonical representation of $G$ and $\hat{G}$ shows that GLWE is equivalent to the fractional version of Regev's original LWE. If $G = \mathbb{Z}_p$ for some prime $p$, then $\hat{G}$ can be defined by multiplications: $\hat{\mathbf{s}}$ is the homomorphism mapping any $t \in \mathbb{Z}_p$ to $ts/p \mod 1$. Thus, GLWE can be viewed as a randomized version of Boneh-Venkatesan's *Hidden Number Problem* [8]: recover a secret number $s \bmod p$, given approximations of $st_i \bmod p$ for many random integers $t_i$'s. By analogy with LWE (see [37,11]), there is a folklore reduction from (Search) Decisional-GLWE$_{G,\leq\alpha}(\mathcal{S})$ to (Search) Decisional-GLWE$_{G,\alpha}(\mathcal{S})$, respectively.

**Lemma 5.** *(Adapted from [11, Lemma 2.13]) Let $\mathcal{A}$ be an algorithm for Decisional*-GLWE$_{G,m,\alpha}(\mathcal{S})$ *(resp. Search) with advantage at least $\varepsilon > 0$. Then there exists an algorithm $\mathcal{B}$ for Decisional*-GLWE$_{G,m',\leq\alpha}(\mathcal{S})$ *(resp. Search) using oracle access to $\mathcal{A}$ and with advantage $\geq 1/3$, where both $m'$ and its running time are poly$(m, 1/\varepsilon, \log \#G)$.*

*Proof.* (Sketch, see App.B.3 of the full version [17] for a detailed proof). Like in LWE, the basic idea is to add noises in small increments to the distribution obtained from the challenger, and feed it to the oracle solving the Decisional-GLWE$_{G,\alpha}(\mathcal{S})$ (resp. Search) and estimate the behavior of the oracle. □

## 4 Structural Lattice Reduction

### 4.1 Overview

A basic result (following from structure theorems of finitely-generated modules over principal ideal domains) states that for any full-rank sublattice $L$ of a full-rank lattice $\bar{L} \subseteq \mathbb{R}^n$, there is a basis $\bar{B} = (\bar{\mathbf{b}}_1, \ldots, \bar{\mathbf{b}}_n)$ of $\bar{L}$ and integers $q_1 \geq q_2 \geq \cdots \geq q_n \geq 1$ s.t. $B = (q_1\bar{\mathbf{b}}_1, \ldots, q_n\bar{\mathbf{b}}_n)$ is a basis of $L$. The $q_i$'s can be made unique by selecting powers of prime numbers, or by requiring each $q_{i+1}$ to divide $q_i$, in which case $q_1, \ldots, q_n$ are the *elementary divisors* of the pair $(\bar{L}, L)$.

In this section, we introduce a lattice reduction converse, which we call *structural lattice reduction*. Lattice reduction asks to find a short basis of a given full-rank lattice $L \subseteq \mathbb{Z}^n$. In structural lattice reduction, one is further given a finite abelian group $G$ of rank $\leq n$, and wants to find a *short* basis of some overlattice $\bar{L}$ of $L$ such that $\bar{L}/L \simeq G$ effectively. More precisely, given a basis $B$ of a full-rank lattice $L \subseteq \mathbb{Z}^n$, a suitable bound $\sigma > 0$ and integers $q_1 \geq \cdots \geq q_k$ defining $G = \mathbb{Z}_{q_1} \times \cdots \times \mathbb{Z}_{q_k}$, one asks to compute a basis $\bar{B}$ of an overlattice $\bar{L} \supseteq L$ such that $\|\bar{B}^*\| \leq \sigma$ and $B = (q_1\bar{\mathbf{b}}_1, \ldots, q_k\bar{\mathbf{b}}_k, \bar{\mathbf{b}}_{k+1}, \ldots, \bar{\mathbf{b}}_n)$ is a basis of $L$. Interestingly, we do not require the input basis $B$ to have integer or rational coefficients, as long as its Gram-Schmidt coefficients are known with enough precision. Indeed, our structural reduction algorithm can simply focus on finding the rational transformation matrix between $\bar{B}$ and $B$.

Previous worst-case to average-case reductions implicitly used the group $G = \mathbb{Z}_q^n$, thus $\bar{L} = L/q$. Here, finding a basis $\bar{B}$ of $\bar{L}$ with small $\|\bar{B}^*\|$ is the same as finding a basis $B = q\bar{B}$ of $L$ with small $\|B^*\|$, which is just lattice reduction. However, we obtain new problems and applications by considering different choices of $G$. In the trivial case $G = \mathbb{Z}_q^n$, $\bar{B} = q^{-1}B$ implies that $\|\bar{B}^*\| = \|B^*\|/q$ where the factor $q$ is exactly $\#G^{1/n}$: this suggests that in general, we might hope to reduce $\|\bar{B}^*\|$ by a factor close to $\#G^{1/n}$, compared to $\|B^*\|$.

Another trivial case of structural lattice reduction is $G = \mathbb{Z}_{q_1} \times \cdots \times \mathbb{Z}_{q_n}$ where the $q_i$'s are distinct positive integers of similar bit-length. If $B = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ is a basis of $L \subseteq \mathbb{Z}^n$, then $\bar{B} = (q_1^{-1}\mathbf{b}_1, \ldots, q_n^{-1}\mathbf{b}_n)$ generates an overlattice $\bar{L}$ such that $\bar{B}^* = (q_1^{-1}\mathbf{b}_1^*, \ldots, q_n^{-1}\mathbf{b}_n^*)$, and therefore $\|\bar{B}^*\| \leq \|B^*\|/\min_{i=1}^n q_i$. The factor $\min_{i=1}^n q_i$ is close to $\#G^{1/n}$ if the $q_i$'s have similar bit-length. But if the $q_i$'s are unbalanced, such as when $\min_{i=1}^n q_i = 1$, then the bound is much weaker. In particular, the case $G = \mathbb{Z}_p$ for some large prime $p$ looks challenging, as the trivial choice $\bar{B} = (p^{-1}\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n)$ looks useless: $\bar{L}/L \simeq G$ but $\|\bar{B}^*\|$ is likely to be essentially as big as $\|B^*\|$, because for a typical reduced basis, the first $\|\mathbf{b}_i^*\|$'s have the same size.

### 4.2 Co-cyclic Lattice Reduction

As a warm-up, we solve structural lattice reduction when the target group $G$ is cyclic of order $q$, which we call *co-cyclic lattice reduction*. Let $\bar{B}$ be a solution of structural reduction on $(L(B), G, \sigma)$: $C = (q\bar{b}_1, \bar{b}_2, \ldots, \bar{b}_n)$ is a basis of $L$ s.t. $\|\mathbf{c}_1\| \leq q\sigma$ and $\|\mathbf{c}_i^*\| \leq \sigma$ for all $i \geq 2$.

**Algorithm 1** Unbalanced Reduction

**Input:** an $n \times m$ basis $B$ of an integer lattice $L \subseteq \mathbb{Z}^m$ and a target length $\sigma \in \mathbb{Q}^+$. More generally, $B$ can be any $n$-dimensional projected block $B = B'_{[i,i+n-1]}$ of some basis $B'$ of $L \subseteq \mathbb{Z}^m$.
**Output:** an $n \times n$ unimodular matrix $U$ such that $C = UB$ satisfies $\|c_i^*\| \leq \sigma$ for $i \geq 2$ and $\|\mathbf{c}_1\| \leq n\sigma\delta_\sigma(B)$.

$C \leftarrow B$, $U \leftarrow I_n$ and compute the Gram-Schmidt matrices $\mu$ and $C^*$
If $\|\mathbf{c}_i^*\| \leq \sigma$ for all $i$, **return** $U$
**for** $i = k - 1$ downto 1 where $k$ is the largest index such that $\|\mathbf{c}_k^*\| > \sigma$ **do**
  **if** $\|\mathbf{c}_i^*\| \leq \sigma$ **then**
    $\alpha \leftarrow \lfloor -\mu_{i+1,i} \rceil$
  **else**
    $\alpha \leftarrow \left\lceil -\mu_{i+1,i} + \frac{\|\mathbf{c}_{i+1}^*\|}{\|\mathbf{c}_i^*\|}\sqrt{(\|\mathbf{c}_i^*\|/\sigma)^2 - 1} \right\rceil$
  **end if**
  $(\mathbf{c}_i, \mathbf{c}_{i+1}) \leftarrow (\mathbf{c}_{i+1} + \alpha \cdot \mathbf{c}_i,\ \mathbf{c}_i)$, $(\mathbf{u}_i, \mathbf{u}_{i+1}) \leftarrow (\mathbf{u}_{i+1} + \alpha \cdot \mathbf{u}_i,\ \mathbf{u}_i)$ and update the GS matrices $\mu$ and $C^*$.
**end for**
**return** $U$

To find such a basis $\bar{B}$, we first show how to transform $B$ to ensure $\|\mathbf{b}_i^*\| \leq \sigma$ for all $i \geq 2$, using a poly-time algorithm which we call *unbalanced reduction* (see Alg. 6). This algorithm can be explained as follows: in dimension two, it is easy to make $\mathbf{b}_2^*$ arbitrarily short by lengthening $\mathbf{b}_1$ (adding a suitable multiple of $\mathbf{b}_2$), since $\|\mathbf{b}_1\| \times \|\mathbf{b}_2^*\| = \mathrm{vol}(L)$ is invariant. Unbalanced reduction works

by iterating this process on two-dimensional projected lattices, similarly to the classical size-reduction process. However, one would like to make sure that the resulting first basis vector $\mathbf{c}_1$ does not become too large, as follows:

**Theorem 2 (Unbalanced reduction).** *Given an $n$-dim projected block $B = B'_{[i,i+n-1]}$ of a lattice $L \subseteq \mathbb{Z}^m$ and a target $\sigma \in \mathbb{Q}^+$, Alg. 6 outputs in polynomial time an $n \times n$ unimodular matrix $U$ such that $C = UB$ satisfies $\|\mathbf{c}_1\| \leq n\sigma\delta_\sigma(B)$ and $\|\mathbf{c}_i^*\| \leq \sigma$ for $i \geq 2$, and:*

$$\delta_\nu(B) \;\leq\; \delta_\nu(C) \;\leq\; \frac{\|\mathbf{c}_1\|}{\sigma\delta_\sigma(B)} \times \delta_\nu(B) \;\text{ for all } \nu \leq \sigma \tag{1}$$

$$\text{where } \delta_\sigma(B) \underset{def}{=} \prod_{j=1}^{n} \max\left(1,\; \left\|\mathbf{b}_j^*\right\|/\sigma\right). \tag{2}$$

We call $\delta_\sigma(B)$ the *cubicity-defect* of $B$ relatively to $\sigma$: it basically measures by which amount the hypercube of side $\sigma$ should be scaled up to cover the parallelepiped spanned by $\mathbf{b}_1^*, \ldots, \mathbf{b}_n^*$. Alg. 6 can be found in App. A.2. The proofs of Th. 2 and Alg. 6 can be found in App.C.2 of the full version of the paper [17]. Th. 2 shows that Alg. 6 solves co-cyclic lattice reduction for $q \geq n\delta_\sigma(B)$. However, this may not be suitable for our applications, since this lower bound depends on $B$ and might be unbounded. To address this issue, we now show that LLL can bound $\delta_\sigma(B)$ depending only on $n$ for appropriate $\sigma$:

**Theorem 3 (LLL's cubicity-defect).** *Let $L$ be a full-rank lattice in $\mathbb{R}^n$ and $\sigma \geq ((1+\varepsilon_{LLL})\sqrt{4/3})^r \cdot \mathrm{bl}(L)$ for some $r \geq 0$. If $B$ is an LLL-reduced basis of $L$ with factor $\varepsilon_{LLL}$, then $\delta_\sigma(B) \leq ((1+\varepsilon_{LLL})\sqrt{4/3})^{\frac{(n-2r)^2}{8}+\frac{(n-2r)}{4}}$.*

By combining Th. 2 and 3, we obtain:

**Theorem 4 (Co-cyclic Reduction).** *Given an $n \times m$ basis of a lattice $L \subseteq \mathbb{Z}^m$, $\varepsilon > 0$ and a rational $\sigma \geq ((1+\varepsilon_{LLL})\sqrt{4/3})^r \cdot \mathrm{bl}(L)$ for some $r \geq 0$, and an integer $q \geq n((1+\varepsilon_{LLL})\sqrt{4/3})^{\frac{(n-2r)^2}{8}+\frac{(n-2r)}{4}}$, Alg. 2 computes a basis $\bar{B}$ of an overlattice $\bar{L} \supseteq L$ in time polynomial in the basis size, $\sigma$ and $1/\varepsilon$, such that $\left\|\bar{B}^*\right\| \leq \sigma$ and $(q\bar{\mathbf{b}}_1, \bar{\mathbf{b}}_2, \ldots, \bar{\mathbf{b}}_n)$ is a basis of $L$. In particular, $\bar{L}/L \simeq \mathbb{Z}_q$.*

For instance, Th. 4 with $r = n$ implies that given a lattice $L$ and any cyclic group $G$ of sufficiently large order $2^{\Omega(n^2)}$, one can efficiently obtain a basis $\bar{B}$ of some overlattice $\bar{L}$ of $L$ such that $\bar{L}/L \simeq G$ and $\|\bar{B}^*\| \leq \mathrm{bl}(L)$: by comparison, an LLL-reduced basis only approximates $\mathrm{bl}(L)$ to some exponential factor.

### 4.3 Arbitrary Groups

Using unbalanced reduction, we prove that for an arbitrary sufficiently large finite abelian group $G$ of rank $\leq n$, given any basis $B$ of the lattice $L \subseteq \mathbb{Z}^n$, one can compute a basis $\bar{B}$ of some overlattice $\bar{L}$ of $L$ s.t. $\bar{L}/L \simeq G$ effectively and $\|\bar{B}^*\|$ is essentially lower than $\|B^*\|/\#G^{1/n}$. In particular, $\mathrm{bl}(\bar{L})$ is essentially

**Algorithm 2** Co-cyclic Reduction

**Input:** a basis of a full-rank integer lattice $L \subseteq \mathbb{Z}^n$, a factor $\varepsilon > 0$, and a rational $\sigma \geq ((1+\varepsilon_{\text{LLL}})\sqrt{4/3})^r \cdot \text{bl}(L)$ for some $r \geq 0$, and an integer $q \geq n((1+\varepsilon_{\text{LLL}})\sqrt{4/3})^{\frac{(n-2r)^2}{8}+\frac{(n-2r)}{4}}$

**Output:** a basis $\bar{B}$ of an overlattice $\bar{L}$ such that $\|\bar{B}^*\| \leq \sigma$ and $\bar{L}/L \simeq \mathbb{Z}_q$.

1: Apply Alg. 6 on an LLL-reduced basis with factor $\varepsilon_{\text{LLL}}$ output by the LLL algorithm.
2: **return** $\bar{B} = (\frac{\mathbf{c}_1}{q}, \mathbf{c}_2, \ldots, \mathbf{c}_n)$ where $C$ is the basis of $L$ returned by Alg. 6.

**Algorithm 3** Structural Lattice Reduction

**Input:** $\sigma$, an $n \times m$ basis $B$ of an integer lattice $L$, and $(q_1, \ldots, q_k)$ s.t. $G = \prod_{i=1}^k \mathbb{Z}_{q_i}$ satisfies the conditions of Th. 5

**Output:** an $n \times m$ basis $\bar{B}$ of an overlattice $\bar{L}$ of $L$ such that $\|\bar{B}^*\| \leq \sigma$ and $\bar{L}/L \simeq G$.

1: $C \leftarrow B$
2: **for** $i = 1$ to $k$ **do**
3: **if** $\|C^*_{[i,n]}\| \leq \sigma$ **return** $\bar{B} = (\frac{\mathbf{c}_1}{q_1}, \ldots, \frac{\mathbf{c}_k}{q_k}, \mathbf{c}_{k+1}, \ldots, \mathbf{c}_n)$
4: Compute the smallest $\ell \geq \sigma$ such that $\ell \cdot \delta_\ell(C_{[i,n]}) = q_i\sigma/(n-i+1)$.
5: $V \leftarrow \text{UnbalancedReduction}(C_{[i,n]}, \sigma)$ using Alg. 6.
6: Apply $V$ on $(\mathbf{c}_i, \ldots, \mathbf{c}_n)$
7: **end for**
8: **return** $\bar{B} = (\frac{\mathbf{c}_1}{q_1}, \ldots, \frac{\mathbf{c}_k}{q_k}, \mathbf{c}_{k+1}, \ldots, \mathbf{c}_n)$

$\#G^{1/n}$ smaller than $\text{bl}(L)$. Although this is slightly weaker than the result we obtained (in the previous subsection) for cyclic groups $G$, it is sufficient for our worst-case to average-case reductions.

**Theorem 5 (Structural Lattice Reduction).** *Given an $n \times m$ basis $B$ of a lattice $L \subseteq \mathbb{Z}^n$, and $k \leq n$ integers $q_1 \geq \cdots \geq q_k$ defining the group $G = \prod_{i=1}^k \mathbb{Z}_{q_i}$ s.t. $n^k(\|B^*\|/\sigma)^n \leq \#G$ or:*
*$\#G \geq \frac{n!}{(n-k)!}\delta_\sigma(B)$ and for all $i \leq k$, $\|B^*\|/\sigma \leq q_i/(n+1-i)$*

*Alg. 3 outputs in polynomial time in $n, m, \|B\|, \log(q_i)$, a basis $\bar{B}$ of an overlattice $\bar{L} \supseteq L$ such that $\|\bar{B}^*\| \leq \sigma$ and $(q_1\bar{\mathbf{b}}_1, \ldots, q_n\bar{\mathbf{b}}_n)$ is a basis of $L$ where $q_i = 1$ for $i > k$. In particular, $\bar{L}/L \simeq G$.*

For instance, the condition $n^k(\|B^*\|/\sigma)^n \leq \#G$ in Th. 5 means that $\sigma$ (and therefore $\|\bar{B}^*\|$) can be chosen as low as $n^{k/n}\|B^*\|/(\#G)^{1/n}$. The proof of Th. 5 can be found in App. C.3 of the full version [17]. Intuitively, Alg. 3 simply applies unbalanced reduction iteratively, cycle by cycle of $G$.

### 4.4 Application

Structural reduction finds a short overlattice basis, which can be used to sample short (overlattice) vectors, and provides effective isomorphisms:

**Proposition 2** *Let $L$ and $\bar{L}$ be two full-rank lattices such that $\bar{L} \supseteq L$ and $\bar{L}/L \simeq G$ where $G$ is an explicit finite abelian group. Given bases $B$ and $\bar{B}$ of resp. $L$ and $\bar{L}$, one can compute in polynomial time a surjective morphism $\varphi$ from $\bar{L}$ to $G$ s.t. $\ker\varphi = L$ (*i.e. $\bar{L}/L \overset{\varphi}{\simeq} G$*), and a "dual" morphism $\varphi^\times : L^\times \to \hat{G}$ s.t.*

$$[\varphi^\times(\mathbf{u})](\varphi(\mathbf{v})) = \langle \mathbf{u}, \mathbf{v} \rangle \mod 1 \text{ for all } \mathbf{u} \in L^\times \text{ and all } \mathbf{v} \in \bar{L} \tag{3}$$

*Furthermore, preimages of $\varphi^\times$ can be computed in polynomial time.*

# 5 Hardness of Group-SIS

Our result requires that the finite abelian group $G$ is *explicit* (see Sect. 2).

## 5.1 Overview

The main idea behind the SIS reduction can be traced back to Mordell's arrithmetical proof [29] of Minkowski's theorem. To prove the existence of short vectors in a full-rank lattice $L \subseteq \mathbb{R}^n$, Mordell implicitly presented an algorithm to find short vectors from (exponentially many) long vectors, as follows. Let $q \geq 1$ be an integer and $\mathbf{w}_1, \ldots, \mathbf{w}_m \in L$ be distinct of norm $\leq R$ where $m > q^n$: for large $R$, $m$ can be as large as the volume of the $R$-radius ball divided by the volume of $L$. Let $\mathbf{v}_i = q^{-1}\mathbf{w}_i \in q^{-1}L$. Since $m > q^n = [(q^{-1}L) : L]$, there are $i \neq j$ such that $\mathbf{v}_i \equiv \mathbf{v}_j \bmod L$, *i.e.* $\mathbf{v}_i - \mathbf{v}_j = q^{-1}(\mathbf{w}_i - \mathbf{w}_j) \in L$ whose (nonzero) norm is $\leq 2R/q$, which is short for appropriate choices of $q$ and $R$.

This algorithm is not efficient since $m$ is exponential in $q$, but it can be made polynomial by reducing $m$ to $\mathrm{poly}(n)$, using a $\mathrm{SIS}(m, n, q)$ oracle. Indeed, let $L$ be a full-rank integer lattice in $\mathbb{Z}^n$. The lattice $\bar{L} = q^{-1}L$ is an overgroup of $L$ such that $\bar{L}/L \simeq \mathbb{Z}_q^n = G$ explicitly: there is an efficiently computable surjective morphism $\varphi : \bar{L} \to G$ s.t. $L = \ker \varphi$, *e.g.* for any basis $(\bar{\mathbf{b}}_1, \ldots, \bar{\mathbf{b}}_n)$ of $\bar{L}$, let $\varphi(\sum_{i=1}^n x_i \bar{\mathbf{b}}_i) = (x_1 \bmod q, \ldots, x_n \bmod q) \in G$.

Furthermore, if $\bar{B}$ is short enough compared to the minima of $L$, it is possible to sample short vectors $\mathbf{v}_1, \ldots, \mathbf{v}_m \in \bar{L}$ with Gaussian distribution of parameter as small as $\eta_\varepsilon(L)$. Fourier analysis guarantees that for such Gaussian distributions, each projection $g_i = \varphi(\mathbf{v}_i)$ is uniformly distributed over $G$. This allows us to call an SIS oracle on $(g_1, \ldots, g_m)$, which outputs a short $\mathbf{x} \in \mathbb{Z}^m$ such that $\sum_{i=1}^m x_i g_i = 0$, *i.e.* $\sum_{i=1}^m x_i \varphi(\mathbf{v}_i) = 0$ which implies that $\mathbf{v} = \sum_{i=1}^m x_i \mathbf{v}_i \in L$. This $\mathbf{v}$ is provably non-zero with overwhelming probability, and is short because the $\mathbf{v}_i$'s and $\mathbf{x}$ are, which concludes the reduction from worst-case SIVP to SIS.

With this formalization, we can replace the SIS oracle by a GSIS oracle if we are able to sample short vectors $\mathbf{v}_1, \ldots, \mathbf{v}_m \in \bar{L}$ with Gaussian distribution, where $\bar{L}/L \simeq G$. And this is exactly what structural lattice reduction ensures. Previous SIS reductions used special choices of $\bar{L}$ and sampled differently short vectors in the overlattice: see App.H. of the full version [17] for a comparision with previous works.

## 5.2 Reducing Worst-case ApproxSIVP to GSIS

Our main result formalizes the previous sketch and states that for appropriate choices of $(G, m, \beta)$, if one can solve $\mathrm{GSIS}(G, m, \beta)$ on average, then one can approximate SIVP in the worst case, *i.e.* one can efficiently find short vectors in every $n$-dimensional lattice:

**Theorem 6.** *Let $n \in \mathbb{N}$ and $\varepsilon = \mathrm{negl}(n)$. Given as input a basis $B$ of a full-rank integer lattice $L \subseteq \mathbb{Z}^n$ and $\sigma \geq \sqrt{2}\,\mathrm{bl}(L)$, and an explicit finite abelian group $G$ of rank $k \leq n$ such that $\#G \geq n^k(\|B^*\|/\sigma)^n$, Alg. 4 outputs (in random poly-time)*

**Algorithm 4** Reducing ApproxSIVP to GSIS

**Input:** a basis $B$ of a full-rank integer lattice $L \in \mathbb{Z}^n$, a parameter $\sigma \geq \sqrt{2}\,\mathrm{bl}(L)$, a negl. $\varepsilon > 0$, an explicit finite abelian group $G$ satisfying the condition of Th. 6, and an oracle $\mathcal{O}$ solving $\mathrm{GSIS}(G, m, \beta)$ with probability $\geq 1/\mathrm{poly}(n)$.
**Output:** A set $S$ of $n$ linearly independent vectors of $L$ of norm $\leq \sigma\eta_\varepsilon(\mathbb{Z}^n)\sqrt{n/2\pi}\beta$.
1: $S \leftarrow \emptyset$.
2: Call structural reduction (Alg. 3) on $(B, G, \sigma)$ to get $\bar{B}$ s.t. $\|\bar{B}^*\| \leq \sigma$ and $\varphi : \bar{L} \to G$ (Prop. 2) where $\bar{L} = L(\bar{B})$.
3: **repeat**
4: Sample $\mathbf{v}_1, \cdots, \mathbf{v}_m \in \bar{L}$ with distribution $D_{\bar{L}, \sigma\eta_\varepsilon(\mathbb{Z}^n), \mathbf{0}}$ using $\bar{B}$.
5: $g_i = \varphi(\mathbf{v}_i)$ for $1 \leq i \leq m$, forming a sequence $\mathbf{g} = (g_1, \ldots, g_m) \in G^m$.
6: Call the GSIS-oracle $\mathcal{O}$ on $\mathbf{g}$, which returns $\mathbf{x} = (x_1, \ldots, x_m) \in \mathbb{Z}^m$ s.t. $\sum_{i=1}^m x_i g_i = 0$.
7: $\mathbf{v} \leftarrow \sum_{i=1}^m x_i \mathbf{v}_i \in L$
8: **if** $\|\mathbf{v}\| \leq \sigma\eta_\varepsilon(\mathbb{Z}^n)\sqrt{n\pi}\beta$ and $\mathbf{v} \notin \mathrm{span}(S)$ **then** $S \leftarrow S \cup \{\mathbf{v}\}$
9: **until** $\dim(S) = n$
10: Return $S$

*$n$ linearly independent vectors of $L$ with norm $\leq \sigma\eta_\varepsilon(\mathbb{Z}^n)\sqrt{n\pi}\beta$, using polynomially many calls to an oracle solving* $\mathrm{GSIS}(G, m, \beta)$ *with prob.* $\geq 1/\mathrm{poly}(n)$.

In particular, letting $\sigma = \frac{\|B\|^*}{2\eta_\varepsilon(\mathbb{Z}^n)\sqrt{n/\pi}\beta}$ gives an incremental version of the reduction, where the output basis is twice as short as the input. This generalizes [28, Th. 5.9] and [20, Th. 9.2] with a GSIS oracle instead of SIS. Iterating Th. 6 until $\sigma = \sqrt{2}\,\mathrm{bl}(L)$ connects GSIS to worst-case ApproxSIVP.

**Corollary 1.** *Let $n \in \mathbb{N}$ and $\varepsilon = \mathrm{negl}(n)$. Let $(G_n)_{n\in\mathbb{N}}$ be a sequence of explicit finite abelian groups of rank $k_n$ s.t. $\#G_n \leq (\beta_n/\sqrt{m_n})^{m_n}$ for $m_n \in \mathbb{N}$. If $\#G_n \geq n^{k_n}\left(\eta_\varepsilon(\mathbb{Z}^n)\sqrt{2n/\pi}\beta_n\right)^{\max(n,k_n)}$, then using polynomially many calls to an oracle solving* $\mathrm{GSIS}(G_n, m_n, \beta_n)$ *with prob.* $\geq 1/\mathrm{poly}(n)$*, one can solve worst-case $n$-dimensional* $\mathrm{ApproxSIVP}_{\eta_\varepsilon(\mathbb{Z}^n)\sqrt{n/\pi}\beta_n}$ *in (randomized) poly-time.*

Consider the set of all full-rank integer lattices $\subseteq \mathbb{Z}^m$ of volume $\geq \omega_n = n^m\left(\eta_\varepsilon(\mathbb{Z}^n)\sqrt{2n/\pi}\beta_n\right)^m$. This set can be partitioned as $\cup_G \mathcal{L}_{G,m}$ where $G$ runs over all finite abelian groups of order $\geq \omega_n$ and rank $\leq m$. Each such $G$ satisfies the conditions of Cor. 1, and therefore GSIS over $G$ is as hard as worst-case lattice problems: for any partition cell $\mathcal{L}_{G,m}$, finding short vectors in a random lattice from this cell is as hard as finding short vectors in any $n$-dim lattice.

## 6 Hardness of Decisional-Group-LWE

We transfer the following Decisional-LWE hardness results to Decisional-GLWE:

**Theorem 7 ([37,34]).** *Let $n \in \mathbb{N}$, $q_n \geq 1$ be a sequence of integers, and $\alpha_n \in (0, 1)$ be a real sequence s.t. $\alpha_n q_n \geq 2\sqrt{n}$. There exists a quantum reduction from worst-case $n$-dimensional* $\mathrm{GapSVP}_{\tilde{O}(n/\alpha_n)}$ *to Decisional-*$\mathrm{GLWE}_{\mathbb{Z}_{q_n}^n, \alpha_n}$*. If $q_n \geq 2^{n/2}$ is smooth then there is a classical reduction between them.*

**Theorem 8 ([11]).** *Let $n \in \mathbb{N}$ and $q_n \geq 1$ be a sequence of integers, and let $\alpha_n \in (0,1)$ be a real sequence such that $\alpha_n \geq 2n^{1/4}/2^{\sqrt{n}/2}$. There exists a classical reduction from worst-case $\sqrt{n}$-dimensional* $\mathrm{GapSVP}_{\tilde{O}(\sqrt{n}/\alpha_n)}$ *to Decisional-*$\mathrm{GLWE}_{\mathbb{Z}_{q_n}^n,\beta_n}$*, where $\beta_n^2 = 10n\alpha_n^2 + \frac{n}{q_n^2} \cdot \omega(\log n)$*

To do so, we reduce Decisional-LWE to Decisional-GLWE using a technique we call group switching. This technique transforms GLWE samples over a group $G$ to another group $G'$, generalizing the modulus-dimension switching technique in [11], which is the special case $G = \mathbb{Z}_q^n$ and $G' = \mathbb{Z}_{q'}^{n'}$. We believe that the group switching technique proposed below is useful to better understand the core idea of the modulus-dimension switching technique.

Before presenting group switching, we note that the modulus-dimension switching technique from [11] implicitly uses a special case of structural lattice reduction. More precisely, Brakerski *et al.* [11] defined a special lattice $\Lambda$ (see Th 3.1 of [11]) to transform LWE samples over $G = \mathbb{Z}_q^n$ to LWE samples over $G' = \mathbb{Z}_{q'}^{n'}$, but the meaning of $\Lambda$ may look a bit mysterious. The lattice $\Lambda$ is defined as $\Lambda = \frac{1}{q'}\mathbb{Z}^{n'} \cdot H + \mathbb{Z}^n$ where $H$ is some $n' \times n$ integer matrix: this matrix is actually denoted by $G$ in [11], but this would collide with our notation $G$ for finite abelian groups. And [11] provided a good basis of $\Lambda$ in special cases. We note that the exact definition of $\Lambda$ is not important: the quotient $\Lambda/\mathbb{Z}^n$ turns out to be isomorphic to the group $G' = \mathbb{Z}_{q'}^{n'}$, as shown by the transformation mapping $\frac{1}{q'}\mathbf{x} \cdot H + \mathbf{y} \in \Lambda$ to $\mathbf{x} \mod q' \in G'$. Thus, finding a good basis of $\Lambda$ is actually a special case of structural lattice reduction for the lattice $\mathbb{Z}^n$ and the group $G'$. Therefore, it is natural to use structural lattice reduction directly (instead of an ad-hoc process) to obtain a more general statement than the modulus-dimension switching technique of [11].

Since we have two groups $G$ and $G'$ and two overlattices $\bar{L}$ and $\bar{L}'$ of $\mathbb{Z}^n$, we have two morphisms $\varphi : \bar{L} \to G$ and $\varphi' : \bar{L}' \to G'$ with $\ker(\varphi) = \ker(\varphi') = \mathbb{Z}^n$. Both morphisms are associated to their dual morphism as in Prop. 2, *i.e.* $\varphi^\times : \mathbb{Z}^n \to \hat{G}$ and $\varphi'^\times : \mathbb{Z}^n \to \hat{G}'$, satisfying $[\varphi'^\times(\mathbf{u})](\varphi'(\mathbf{v})) = \langle \mathbf{u}, \mathbf{v} \rangle \mod 1$ for all $\mathbf{u} \in \mathbb{Z}^n$ and all $\mathbf{v} \in \bar{L}'$ (resp. without primes).

We say that a distribution $S$ over $\mathbb{Z}^n$ is $K$-bounded if $\Pr_{\mathbf{s} \leftarrow S}[\|\mathbf{s}\| > K] \leq \mathrm{negl}(n)$. By extension, given a (public) morphism $f : \mathbb{Z}^n \to \hat{G}$, we say that a distribution $\mathcal{S}$ over $\hat{G}$ is $K$-bounded (for $f$) if it is the image of a $K$-bounded distribution[8] by $f$. In the following, we choose $\varphi^\times = f$ and $\varphi$ its dual morphism accordingly. Thus, any secret $\hat{s} \leftarrow \mathcal{S}$ has with overwhelming probability a preimage $\mathbf{s} \in \mathbb{Z}^n$ of norm $\leq K$. Note that the small $\mathbf{s} \in \mathbb{Z}^n$ may be hard to compute from $\hat{s}$, however what matters is its existence. During group switching, the new secret in $\hat{G}'$ will be $\varphi'^\times(\mathbf{s})$, and the new $K$-bounded distribution $\mathcal{S}' = \varphi'^\times(S)$.

**Lemma 6 (Group Switching).** *Let $G$ and $G'$ be two finite abelian groups of rank $\leq n$ s.t. $G$ is fully-explicit and $G'$ is explicit. Let $\bar{L}$ be an overlattice of $\mathbb{Z}^n$ such that $\bar{L}/\mathbb{Z}^n \simeq G$. Let $\bar{B}'$ be a basis of an overlattice $\bar{L}'$ of*

[8] Ideally, $f$ should be collision resistant among samples from $S$. In the classical LWE ($G = \mathbb{Z}_q^n$), $f$ maps $\mathbf{s} \in \mathbb{Z}^n$ to the secret character $\hat{s} : \mathbf{y} \to {}^1/_q\langle \mathbf{s}, \mathbf{y} \rangle \mod 1$ in $\hat{G}$.

*$\mathbb{Z}^n$ such that $\bar{L}'/\mathbb{Z}^n \simeq G'$. Let $\varphi, \varphi'$ and $\varphi'^{\times}$ be defined as in Prop. 2. Let $r \geq \max\left(\sqrt{2}\eta_\varepsilon(\bar{L}), \|\bar{B}'^*\| \cdot \eta_\varepsilon(\mathbb{Z}^n)\right)$, where $\varepsilon$ is some negligible function. Then, there is an efficient randomized algorithm which, given as input a sample from $G \times \mathbb{T}$, outputs a sample from $G' \times \mathbb{T}$, with the following properties:*
*- If the input sample has uniform distribution in $G \times \mathbb{T}$, then the output sample has uniform distribution in $G' \times \mathbb{T}$ (except with negligible distance).*
*- If the input is distributed according to $A_{G,\alpha}(\hat{s})$ for some $\hat{s} = \varphi^{\times}(\mathbf{s})$ s.t. $\mathbf{s} \in \mathbb{Z}^n$ and $\|\mathbf{s}\| \leq K$, then the output distribution is statistically close to $A_{G',\beta}(\hat{s}')$, where $\hat{s}' = \varphi'^{\times}(\mathbf{s}) \in \hat{G}'$ and $\beta^2 = \alpha^2 + r^2(\|\mathbf{s}\|^2 + K^2) \leq \alpha^2 + 2(rK)^2$.*

By combining Group Switching (Lemma 6) with structural reduction (Th. 5), one derives a reduction between Decisional-GLWE of two groups $G$ and $G'$:

**Corollary 2 (GLWE to GLWE).** *Let $n \in \mathbb{N}$ and $0 < \sigma_n < 1$ be a real sequence. Let $(G_n)_{n\in\mathbb{N}}$ and $(G'_n)_{n\in\mathbb{N}}$ be two sequences of finite abelian groups with respective rank $k_n \leq n$ and $k'_n \leq n$ s.t. $\#G_n \geq n^{k_n}(\sqrt{2}/\sigma_n)^n$ (or if $G_n = \mathbb{Z}_{q_n}^n$ where $q_n \geq \sqrt{2}/\sigma_n$) and $\#G'_n \geq n^{k'_n}(1/\sigma_n)^n$. Assume that $G_n$ is fully-explicit and $G'_n$ is explicit. Let $S$ be an arbitrary $K_n$-bounded distribution over $\mathbb{Z}^n$ and $\mathcal{S} = \varphi^{\times}(S)$ its image by some morphism $\varphi^{\times} : \mathbb{Z}^n \to \hat{G}_n$, $\alpha_n, \beta_n > 0$ be two real sequences and $\varepsilon = \mathrm{negl}(n)$ satisfying $\beta_n^2 \geq \alpha_n^2 + 2(\sigma_n K_n \cdot \eta_\varepsilon(\mathbb{Z}^n))^2$. Then there is an efficient reduction from Decisional-$\mathrm{GLWE}_{G_n,\leq\alpha_n}(\mathcal{S})$ to Decisional-$\mathrm{GLWE}_{G'_n,\leq\beta_n}(\mathcal{S}')$, where $\mathcal{S}' = \varphi'^{\times}(S)$ for some morphism $\varphi'^{\times} : \mathbb{Z}^n \to \hat{G}'_n$*

*Proof.* Given the canonical basis of $\mathbb{Z}^n$ and $G_n$, structural reduction finds an overlattice $\bar{L}$ together with a basis $\bar{C}$ s.t. $\left\|\bar{C}^*\right\| \leq \sigma_n/\sqrt{2}$. Therefore $\sqrt{2}\eta_\varepsilon(\bar{L}) \leq \sigma_n\eta_\varepsilon(\mathbb{Z}^n)$. And structural reduction on $G'_n$ and $\sigma_n$ gives a short basis $\bar{B}'$ of length $\leq \sigma_n$ and defines $\bar{L}'$. The rest follows immediately from Lemma 6. □

Using the normal form [3] of LWE, namely, if $\mathcal{S}$ is the image of the $\alpha_n q_n\sqrt{n}$-bounded distribution $\mathcal{D}_{\mathbb{Z}^n,\alpha_n q_n}$, through the canonical embedding which maps $\mathbf{s} \in \mathbb{Z}^n$ to the character $\hat{s} = \mathbf{y} \to {}^1/_{q_n}\langle \mathbf{s}, \mathbf{y}\rangle \mod 1$, we obtain the quantum/classical hardness of Decisional-GLWE problem for any sufficiently large finite abelian group, together with Theorems 7 and 8:

**Corollary 3 (Quantum Hardness of GLWE).** *Let $n \in \mathbb{N}$ and $q_n \geq 1$ be a sequence of integers and $(G'_n)_{n\in\mathbb{N}}$ be a sequence of any finite abelian explicit groups such that $\#G'_n \geq n^{k_n}(q_n/\sqrt{2})^n$ where $k_n = \mathrm{rank}\,(G'_n) \leq n$. Let $\alpha_n, \beta_n \in (0,1)$ be two real sequences such that $\alpha_n q_n \geq 2\sqrt{n}$ and $\beta_n = \alpha_n\sqrt{n} \cdot \omega(\sqrt{\log n})$. Then there exists a quantum reduction from worst-case $n$-dimensional $\mathrm{GapSVP}_{\tilde{O}(n/\alpha_n)}$ to Decisional-$\mathrm{GLWE}_{G'_n,\beta_n}$.*

The lower bound on $\#G'_n$ is better than the lower bound on $\#G_n$ in Cor 1 and for solving Approx-SIVP using a Search-GLWE oracle (see App. E.2. of the full version [17]), because group switching relies on structural reduction over $\mathbb{Z}^n$ rather than an arbitrary lattice: the canonical basis of $\mathbb{Z}^n$ is orthonormal, which simplifies the bound of Sect. 4.

**Corollary 4 (Classical Hardness of GLWE).** *Let $n \in \mathbb{N}$ and $q_n \geq 1$ be a sequence of integers and $(G'_n)_{n\in\mathbb{N}}$ be a sequence of any finite abelian explicit groups such that $\#G'_n \geq n^{k_n}(q_n/\sqrt{2})^n$ where $k_n = \operatorname{rank}(G'_n) \leq n$. Let $\alpha_n, \beta_n \in (0,1)$ be two real sequences such that $\alpha_n \geq 2n^{1/4}/2^{\sqrt{n}/2}$ and $\beta_n^2 = n^2\alpha_n^2 \cdot \omega(\log n) + \frac{n^2}{q_n^2} \cdot \omega(\log^2 n)$. There exists a classical reduction from worst-case $\sqrt{n}$-dimensional $\mathrm{GapSVP}_{\tilde{O}(\sqrt{n}/\alpha_n)}$ to Decisional-$\mathrm{GLWE}_{G'_n,\beta_n}$.*

# 7 Abstracting Lattice Cryptography: Fully-Homomorphic Encryption from GLWE

We showed that GSIS/GLWE are hard under the same worst-case assumptions as SIS/LWE. This suggests to abstract lattice schemes based on SIS/LWE using an arbitrary finite abelian group $G$, and check that the security proof carries through. This may lead to a better understanding of the scheme and a clearer presentation: lattice schemes are typically described using matrices and vectors, which our abstraction avoids.

We illustrate this approach with fully-homomorphic encryption. First, we introduce a GLWE-based El Gamal-like encryption scheme, which generalizes Regev's LWE-based encryption [37] and its dual version [20]. Next, we extend this GLWE generalization of Regev's encryption into a somewhat-homomorphic encryption, by carefully abstracting the Alperin-Sheriff-Peikert variant [2] of the Gentry-Sahai-Waters homomorphic scheme [21]. In particular, we show how to evaluate any boolean function with a noise overhead proportional to the square root of its number of variables, how to recognize any regular language with a noise overhead proportional to the length of the tested word, and how to bootstrap the whole system with only a linear noise overhead instead of quadratic in [2].

## 7.1 A GLWE Variant of El Gamal Encryption

El Gamal encryption combines the one-time pad with Diffie-Hellman. By analogy, we first present a GLWE variant of DH. We consider a (sufficiently large) finite abelian group $G$ and $\mathbf{g} = (g_1, ..., g_m) \in G^m$ chosen uniformly at random. This defines two one-way functions:

- Let $f_{\mathbf{g}} : \mathbb{Z}^m \to G$ be the morphism defined by $f_{\mathbf{g}}(\mathbf{x}) = \sum_{i=1}^m x_i.g_i$, where $x_i.g_i$ is defined by the $\mathbb{Z}$-module structure of $G$. For suitable input distributions $\mathcal{D}$, such as the uniform distribution over $\{0,1\}^m$ or some well-chosen discrete Gaussian distribution, the distribution of $f_{\mathbf{g}}(\mathbf{x})$ becomes statistically close to uniform (*e.g.* see the left-over-hash lemma), and $f_{\mathbf{g}}$ becomes one-way under GSIS.
- Let $f^{\times}_{\mathbf{g}} : \hat{G} \times \mathbb{T}^m \to \mathbb{T}^m$ defined by $f^{\times}_{\mathbf{g}}(\hat{s}, \mathbf{e}) = (\hat{s}(g_1) + e_1, \ldots, \hat{s}(g_m) + e_m)$: if $\hat{s} \in_R \hat{G}$ and $\mathbf{e}$ is sampled from a suitable distribution such as $\mathcal{D}^m_\alpha$, then inverting $f^{\times}_{\mathbf{g}}(\hat{s}, \mathbf{e})$ is search-GLWE, and distinguishing $f^{\times}_{\mathbf{g}}(\hat{s}, \mathbf{e})$ from random is decisional-GLWE.

Consider the bilinear map $\theta : \hat{G} \times \mathbb{Z}^m \to \mathbb{T}$ defined by $\theta(\hat{s}, \mathbf{x}) = \hat{s}(f_{\mathbf{g}}(\mathbf{x}))$. Then $\theta(\hat{s}, \mathbf{x})$ can be efficiently computed from $(\hat{s}, \mathbf{x})$. But it can be computed knowing only $(\hat{s}, f_{\mathbf{g}}(\mathbf{x}))$, or approximately knowing only $(f_{\mathbf{g}}^{\times}(\hat{s}, \mathbf{e}), \mathbf{x})$ by $\sum_{i=1}^m c_i x_i$ (where $\mathbf{c} = f_{\mathbf{g}}^{\times}(\hat{s}, \mathbf{e})$), provided that $\mathbf{e}$ and $\mathbf{x}$ are sampled from suitable distributions. This motivates a GLWE noisy key exchange where Alice and Bob compute their own approximation of $\theta(\hat{s}, \mathbf{x})$: Alice picks $\mathbf{x} \in \mathbb{Z}^m$ from some suitable distribution $\mathcal{D}$, and discloses $y = f_{\mathbf{g}}(\mathbf{x})$; Bob picks $\hat{s} \in_R \hat{G}$ and $\mathbf{e}$ from the distribution $\mathcal{D}_\alpha^m$, and discloses $\mathbf{c} = f_{\mathbf{g}}^{\times}(\hat{s}, \mathbf{e})$. Alice computes her key as $\sum_{i=1}^m c_i x_i$, and Bob computes his key as $\hat{s}(y) + e$ where $e$ is sampled from $\mathcal{D}_\alpha$. Both keys are close to $\theta(\hat{s}, \mathbf{x})$. But, as opposed to Diffie-Hellman, Alice and Bob do not have symmetric roles, which leads to two El Gamal cryptosystems by swapping Alice and Bob roles: this is why Regev encryption has a so-called dual variant [20]. We now give a detailed description of the main cryptosystem, which generalizes Regev's [37], and which we use in our fully-homomorphic encryption.

Define the group $H = G \times \mathbb{T}_k$ where $k \in \mathbb{N}^+$ and $\mathbb{T}_k = \frac{1}{2^k}\mathbb{Z}/\mathbb{Z} \subseteq \mathbb{T}$ is a discretized torus.

$\texttt{GLWE.Gen}(1^n)$ : Takes as input a security parameter $n$, it chooses a Gaussian parameter $0 < \alpha < 1$, a (sufficiently large) finite abelian group $G$ and $m \in \mathbb{N}$. Choose $\mathbf{g} = (g_1, ..., g_m) \in_R G^m$, $\hat{s} \in_R \hat{G}$ and $m$ Gaussian samples $e_1, ..., e_m \leftarrow \mathcal{D}_\alpha$. Set the public key $pk = (\mathbf{g}, \mathbf{y}) \in G^m \times \mathbb{T}_k^m$, where $y_i = \hat{s}(g_i) + e_i \in \mathbb{T}$, and the secret key $sk = \hat{s}$, *i.e.* $\mathbf{y} = f_{\mathbf{g}}^{\times}(\hat{s}, \mathbf{e})$.

$\texttt{GLWE.Enc}(pk, \mu)$ : Takes as input the public key $pk = (\mathbf{g}, \mathbf{y}) \in G^m \times \mathbb{T}_k^m$ and a message $\mu \in \{0, 1\}$. It selects $\mathbf{x} = (x_1, ..., x_m) \in_R \{0, 1\}^m$, and returns $(d, c) \in H$, where $d = f_{\mathbf{g}}(\mathbf{x}) = \sum_{i=1}^m x_i g_i \in G$ and $c = \sum_{i=1}^m x_i y_i + \mu/2 \in \mathbb{T}_k$. Here, $\sum_{i=1}^m x_i y_i$ is Alice's key in the GLWE key exchange. Both $d$ and $c$ use the $\mathbb{Z}$-module structure of $G$ and $\mathbb{T}_k$.

$\texttt{GLWE.Dec}(sk, (d, c))$ : Returns $\mu = \lfloor 2 \cdot (c - \hat{s}(d)) \rceil \bmod 2$ where $sk = \hat{s}$ and $(d, c) \in H$ is the ciphertext.

One obtains a dual scheme by swapping the two one-way functions $f_{\mathbf{g}}$ and $f_{\mathbf{g}}^{\times}$.

**Lemma 7 (Correctness).** *If $0 < \alpha < 1/(4 \cdot \sqrt{m} \cdot \omega(\sqrt{\log n}))$, the main GLWE public-key encryption scheme decrypts correctly with probability $1 - negl(n)$.*

*Proof.* We have: $c - \hat{s}(d) = \sum_{i=1}^m x_i(\hat{s}(g_i) + e_i) + \mu/2 - \hat{s}(\sum_{i=1}^m x_i g_i) = \mu/2 + \sum_{i=1}^m x_i e_i$. It is sufficient to show $|\sum_{i=1}^m x_i e_i| < 1/4$. Let $w \leq m$ be the Hamming weight of $\mathbf{x}$, we know that $\sum_{i=1}^m x_i e_i$ is distributed as $\mathcal{D}_{\sqrt{w}\alpha}$. Therefore, it implies that $|\sum_{i=1}^m x_i e_i| < \sqrt{w}\alpha \cdot \omega(\sqrt{\log n})$ with probability $1 - \exp(-\pi \cdot \omega(\log n)) = 1 - \mathrm{negl}(n)$. We obtain that $|\sum_{i=1}^m x_i . e_i| < \sqrt{w}\alpha \cdot \omega(\sqrt{\log n}) \leq 1/4$ with probability $1 - \mathrm{negl}(n)$, as desired. □

**Lemma 8 (Security).** *If $m \geq 2(\log \#G + k) + \omega(\log n)$ and the $\mathrm{GLWE}_{G,m,\alpha}$ assumption holds, then the main GLWE public-key encryption scheme is IND-CPA secure.*

*Proof.* $\mathbf{g} \in G^m$ is uniformly distributed. By the $\mathrm{GLWE}_{G,m,\alpha}$ assumption, $\mathbf{y} \in \mathbb{T}_k^m$ is computationally indistinguishable from uniform, hence $(\mathbf{g}, \mathbf{y})$ too. Since

$m \geq 2 \cdot \log \#H + \omega(\log n)$ and $\mathbf{x} \in_R \{0,1\}^m$, the left-over-hash lemma ensures that $\sum_{i=1}^m x_i(g_i, y_i)$ is computationally indistinguishable from uniform over $H$, and hence $(d, c)$ too. This proves IND-CPA security. □

### 7.2 A GLWE Variant of GSW Homomorphic Encryption

We now show how to generalize the AP variant [2] of GSW [21] Homomorphic encryption. Let GLWE$(G, \alpha)$ be a black-box instance of GLWE El Gamal encryption over the GLWE group $G$. All noises are discretized in the torus $\mathbb{T}_k = \frac{1}{2^k}\mathbb{Z}/\mathbb{Z} \subseteq \mathbb{T}$ where $2^k\alpha \approx \eta_\varepsilon(\mathbb{Z})$. The group $H = G \times \mathbb{T}_k$ is of special interest.

First, recall that El Gamal encryption is homomorphic with respect to the group operation. Because GLWE$(G, \alpha)$ is a noisy variant of El Gamal encryption, it is also homomorphic for a bounded number of XOR. More precisely, any GLWE ciphertext of a message $\mu \in \{0,1\}$ can be written as $c_1 + \mu h_1 \in H$, where $c_1 = \sum_{i=1}^m x_i(g_i, y_i) \in H$ is a random ciphertext of 0, and $h_1 = (0, 1/2) \in H$. Here, we use the $\mathbb{Z}$-module structure of $H$. The GLWE secret key $\hat{s}$ induces a homomorphism $\texttt{Phase} : H \to \mathbb{T}$ defined as $\texttt{Phase}((a,b)) = b - \hat{s}(a)$. By definition of GLWE, we have $\texttt{Phase}((g_i, y_i)) \approx 0$ for all $1 \leq i \leq m$, but $\texttt{Phase}(h_1) = 1/2$. It follows that the phase of a GLWE ciphertext of a message $\mu$ is $\approx \mu/2$, which explains the GLWE decryption procedure: a ciphertext of 0 is close to the kernel of the phase, while a ciphertext of 1 is far away. Because Phase is a homomorphism and $h_1$ has order 2 in $H$, if $n$ messages $\mu_1, \ldots, \mu_n \in \{0,1\}$ are GLWE-encrypted, then the sum of these $n$ ciphertexts will de decrypted as $\mu_1 \oplus \cdots \oplus \mu_n$, provided that $n$ is not too large.

To achieve more homomorphic operations, one exploits a special property of lattice problems which is not shared by discrete logarithm problems: with special choices of generators, the SIS one-way function can be inverted. To do so, one first extends $h_1$ into a generating set of the $\mathbb{Z}$-module $H$: let $h_2, \ldots, h_\ell \in H$ be such that $\mathbf{h} = (h_1, \ldots, h_\ell)$ is a generating set of $H$. Recall that the GSIS function $f_{\mathbf{g}}$ from Sect.7.1 can be defined over any group: here, we use $H$, so $f_{\mathbf{h}}(\mathbf{x}) = \sum_{i=1}^\ell x_i h_i \in H$ for $(x_1, \ldots, x_\ell) \in \mathbb{Z}^\ell$. Since $\mathbf{h}$ generates $H$, $f_{\mathbf{h}}$ is surjective, and thus, admits a pseudo-inverse $f_{\mathbf{h}}^{-1}$ from $H$ to $\mathbb{Z}^\ell$, such that $f_{\mathbf{h}}(f_{\mathbf{h}}^{-1}(b)) = b$ for any $b \in H$. We also define $F_{\mathbf{h}} : \mathbb{Z}^{\ell \times \ell} \to H^\ell$ by $F_{\mathbf{h}}(\mathbf{X}) = (f_{\mathbf{h}}(\mathbf{x}_1), ..., f_{\mathbf{h}}(\mathbf{x}_\ell))$, where $\mathbf{x}_i$ is the $i$-th row of $\mathbf{X}$. Accordingly, we define $F_{\mathbf{h}}^{-1} : H^\ell \to \mathbb{Z}^{\ell \times \ell}$.

Given a target in $H$, finding a short $f_{\mathbf{h}}()$-preimage corresponds to the GSIS problem, which is in general hard, but it becomes easy for special choices of $\mathbf{h}$, like super-increasing knapsacks: following [26], we call *gadget* such a $\mathbf{h}$. We say that $f_{\mathbf{h}}^{-1}()$ is $\beta$-bounded for $\mathbf{h}$, if $\left\|f_{\mathbf{h}}^{-1}(b)\right\|_\infty \leq \beta \in \mathbb{R}^+$ for any $b \in H$. For instance, if the group $G$ is $\mathbb{Z}_N$ where $2^p < N < 2^{p+1}$, a suitable gadget is $\mathbf{h} = ((0, \frac{1}{2}), (0, \frac{1}{4}), \ldots, (0, \frac{1}{2^k}), (1,0), (2,0), \ldots, (2^p, 0))$, $f_{\mathbf{h}}^{-1}() \in \{0,1\}^\ell$ can be computed by binary decomposition and is 1-bounded for $\mathbf{h}$. This construction can easily be generalized to any fully-explicit $G$, using component-wise binary decomposition: if $G = \mathbb{Z}_q^n$, this corresponds to the Flatten/BitDecomp algorithms proposed in [21] and [2]. However, other algorithms are possible, such as ternary decompositions with preimages in $\{0, \pm 1\}^\ell$.

Given the GLWE encryption scheme $(\texttt{GLWE.Gen}, \texttt{GLWE.Enc}, \texttt{GLWE.Dec})$ described in Sect. 7.1 as a "black box", we build homomorphic encryption using a gadget $\mathbf{h} \in H^\ell$ whose first element is $(0, \frac{1}{2})$:

$\texttt{GSW.Gen}(1^n)$ : Takes as input a security parameter $n$, it runs the key generation algorithm $(pk, sk) \leftarrow \texttt{Gen}(1^n)$, where $pk = (\mathbf{g}, \mathbf{y}) \in G^m \times \mathbb{T}_k^m$ and $sk = \hat{s} \in \hat{G}$.

$\texttt{GSW.Enc}(pk, \mu)$ : Takes as input the public key $pk \in G^m \times \mathbb{T}_k^m$ and a message $\mu \in \{0, 1\}$, it first generates $\ell$ ciphertexts $c_1 = \texttt{GLWE.Enc}(pk, 0), ..., c_\ell = \texttt{GLWE.Enc}(pk, 0)$ of zero, and returns $\mathbf{c} = (c_1, ..., c_\ell) + \mu \cdot \mathbf{h} \in H^\ell$.
This is reminiscent of the GLWE scheme, where a GLWE-ciphertext of a message $\mu$ is of the form $c_1 + \mu h_1 \in H$ where $c_1$ is a random GLWE-ciphertext of 0. Because the first entry of $\mathbf{h}$ is $(0, \frac{1}{2})$, the first entry of $\mathbf{c}$ is a GLWE encryption of $\mu$.

$\texttt{GSW.Dec}(sk, \mathbf{c})$ : Returns $\texttt{GLWE.Dec}(\hat{s}, c_1)$ where $sk = \hat{s}$ and $c_1 \in H$ is the first entry of $\mathbf{c}$.

The security of the scheme and the correctness of decryption follow from that of the GLWE cryptosystem:

**Lemma 9.** *Suppose* $(\texttt{Gen}, \texttt{Enc}, \texttt{Dec})$ *uses samples from* $GLWE_{G,m,\alpha}$. *If* $m \geq 2(\log \#G + k) + \omega(\log n)$ *and* $0 < \alpha < 1/(4 \cdot \sqrt{m} \cdot \omega(\sqrt{\log n}))$, $(\texttt{GSW.Gen}, \texttt{GSW.Enc}, \texttt{GSW.Dec})$ *is IND-CPA secure under the* $GLWE_{G,m,\alpha}$ *assumption, and* $\texttt{GSW.Dec}$ *decrypts correctly with probability* $1 - \texttt{negl}(\lambda)$.

*Proof.* The proof of IND-CPA security is similar to Lemma 8. Since the first entry of $\mathbf{c}$ is a ciphertext of $\mu$ under $\hat{s}$ of the scheme $(\texttt{Gen}, \texttt{Enc}, \texttt{Dec})$, correctness follows from Lemma 7. □

We now describe our homomorphic operations on ciphertexts, namely how to encode Not, And, and Mux gates. First, we note that the GSW-GLWE scheme inherits the $\oplus$-homomorphic properties of the GLWE scheme. Any circuit can be built using only Not and And elementary gates. We chose to add the Mux ternary gate, which encodes the conditional operator $\texttt{Mux}(a, b, c) = a?b{:}c$, because resulting circuits are smaller than NAND-only circuits, all binary gates can be encoded by a single Mux (and a few Not), and it is trivial to batch-convert any truth-table to its corresponding Mux-based binary decision diagram.

**Definition 2 (Homomorphic operations).** *For all ciphertexts* $\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3 \in H^\ell$, *we define:*

$$\texttt{GSW.Not}(\mathbf{c}_1) = \mathbf{h} - \mathbf{c}_1, \quad \texttt{GSW.And}(\mathbf{c}_1, \mathbf{c}_2) = F_{\mathbf{c}_1}\Big(F_{\mathbf{h}}^{-1}(\mathbf{c}_2)\Big),$$
$$\texttt{GSW.Mux}(\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3) = F_{\mathbf{c}_1}\Big(F_{\mathbf{h}}^{-1}(\mathbf{c}_2)\Big) + F_{\mathbf{h}-\mathbf{c}_1}\Big(F_{\mathbf{h}}^{-1}(\mathbf{c}_3)\Big)$$

We express $\texttt{Xor}(a, b)$ as $\texttt{Mux}(a, \texttt{Not}(b), b)$. We naturally extend the Phase homomorphism to $H^\ell$ as $\texttt{Phase} : H^\ell \to \mathbb{T}^\ell$ defined as $\texttt{Phase}(\mathbf{z}) = (b_1 - \hat{s}(a_1), \ldots, b_\ell - \hat{s}(a_\ell)) \in \mathbb{T}^\ell$ where $\mathbf{z} = ((a_1, b_1), \ldots, (a_\ell, b_\ell)) \in H^\ell$. Note that a valid ciphertext of a bit $\mu$ is of the form $\mathbf{c} = \mathbf{z} + \mu\mathbf{h}$ where its *homogeneous*

part $\mathbf{z}$ has a small phase. This small $\mathtt{Phase}(\mathbf{z}) = \mathtt{Phase}(\mathbf{c} - \mathtt{GSW.Dec}(\mathbf{c}).\mathbf{h}) \in \mathbb{T}^\ell$ will be denoted by $\mathtt{Noise}(\mathbf{c})$.

By definition, the decryption function will successfully decrypt any ciphertext $\mathbf{c} \in H^\ell$ such that $\|\mathtt{Noise}(\mathbf{c})\|_\infty < \frac{1}{4}$, where the max-norm in $\mathbb{T}^\ell$ is taken over all coordinates centered in the interval $(-\frac{1}{2}, \frac{1}{2}]$. This is of course the case of fresh GSW.GLWE ciphertexts, whose Gaussian noise has small parameter $\alpha$.

We now show that the GSW.Not, GSW.And and GSW.Mux gates amplify the noise only by a small factor if $f_{\mathbf{h}}^{-1}()$ is $\beta$-bounded.

**Lemma 10 (Worst-case noise of primitive gates).** *Suppose $f_{\mathbf{h}}^{-1}()$ is $\beta$-bounded for some $\beta \in \mathbb{R}^+$. Let $\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3 \in H^\ell$ be three ciphertexts such that $\mathbf{c}_1 = \mathbf{z}_1 + \mu_1 \cdot \mathbf{h}$, $\mathbf{c}_2 = \mathbf{z}_2 + \mu_2 \cdot \mathbf{h}$ and $\mathbf{c}_3 = \mathbf{z}_3 + \mu_3 \cdot \mathbf{h}$, where $\|\mathtt{Phase}(\mathbf{z}_1)\|_\infty \leq B$ and $\|\mathtt{Phase}(\mathbf{z}_2)\|_\infty, \|\mathtt{Phase}(\mathbf{z}_3)\|_\infty < B'$ for some $B, B' \in \mathbb{R}^+$. Then:*

$$\mathtt{GSW.Not}(\mathbf{c}_1) = \mathbf{z} + \mathtt{NOT}(\mu_1) \cdot \mathbf{h} \text{ with } \|\mathtt{Phase}(\mathbf{z})\|_\infty = B \quad (4)$$

$$\mathtt{GSW.And}(\mathbf{c}_1, \mathbf{c}_2) = \mathbf{z}' + (\mu_1 \ \mathtt{AND} \ \mu_2) \cdot \mathbf{h} \text{ with } \|\mathtt{Phase}(\mathbf{z}')\|_\infty \leq \ell\beta B + B' \quad (5)$$

$$\mathtt{GSW.Mux}(\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3) = \mathbf{z}'' + (\mu_1 ? \mu_2 : \mu_3) \cdot \mathbf{h} \text{ with } \|\mathtt{Phase}(\mathbf{z}'')\|_\infty \leq 2\ell\beta B + B' \quad (6)$$

*Proof.* By definition of GSW, we have $\mathtt{GSW.Not}(\mathbf{c}_1) = -\mathbf{z}_1 + \mathtt{NOT}(\mu_1)$, so $\mathbf{z} = -\mathbf{z}_1$, which proves (4). Then,

$$\begin{aligned}\mathtt{GSW.And}(\mathbf{c}_1, \mathbf{c}_2) &= F_{\mathbf{z}_1 + \mu_1 \cdot \mathbf{h}}\Big(F_{\mathbf{h}}^{-1}(\mathbf{c}_2)\Big) = F_{\mathbf{z}_1}(F_{\mathbf{h}}^{-1}(\mathbf{c}_2)) + \mu_1 F_{\mathbf{h}}(F_{\mathbf{h}}^{-1}(\mathbf{c}_2)) \\ &= F_{\mathbf{z}_1}(F_{\mathbf{h}}^{-1}(\mathbf{c}_2)) + \mu_1 \cdot \mathbf{c}_2 = \underbrace{F_{\mathbf{z}_1}(F_{\mathbf{h}}^{-1}(\mathbf{c}_2)) + \mu_1 \mathbf{z}_2}_{\mathbf{z}'} + \mu_1\mu_2 \cdot \mathbf{h}\end{aligned}$$

Letting $\mathbf{z}' = F_{\mathbf{z}_1}(F_{\mathbf{h}}^{-1}(\mathbf{c}_2)) + \mu_1 \mathbf{z}_2$, we have $\mathtt{Phase}(\mathbf{z}') = \mathtt{Phase}(\mathbf{z}_1) \cdot (F_{\mathbf{h}}^{-1}(\mathbf{c}_2))^t + \mu_1 \mathtt{Phase}(\mathbf{z}_2)$, and therefore $\|\mathtt{Phase}(\mathbf{z}')\|_\infty \leq \ell \left\|F_{\mathbf{h}}^{-1}(\mathbf{c}_2)\right\|_\infty \|\mathtt{Phase}(\mathbf{z}_1)\|_\infty + \|\mathtt{Phase}(\mathbf{z}_2)\|_\infty \leq \ell\beta B + B'$, which proves (5). Finally, $\mathtt{GSW.Mux}(\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3)$ is expressed as $\mathtt{GSW.And}(\mathbf{c}_1, \mathbf{c}_2)$ plus $\mathtt{GSW.And}(\mathtt{GSW.Not}(\mathbf{c}_1), \mathbf{c}_3)$. By expanding, the expression takes the form $\mathbf{z}'' + (\mu_2\mu_1 + \mu_3(1-\mu_1)) \cdot \mathbf{h}$ where $\mathbf{z}'' = F_{\mathbf{z}_1}(F_{\mathbf{h}}^{-1}(\mathbf{c}_2)) + F_{\mathbf{z}_1}(F_{\mathbf{h}}^{-1}(\mathbf{c}_3)) + \mu_1\mathbf{z}_2 + (1-\mu_1)\mathbf{z}_3$. Thus, $\mathtt{Phase}(\mathbf{z}'') = \mathtt{Phase}(\mathbf{z}_1) \cdot (F_{\mathbf{h}}^{-1}(\mathbf{c}_2) + F_{\mathbf{h}}^{-1}(\mathbf{c}_3)) + \mu_1 \mathtt{Phase}(\mathbf{z}_2) + (1-\mu_1)\mathtt{Phase}(\mathbf{z}_3)$. The norm of the first term is bounded by $2\ell\beta B$ and among the last two terms, only one is non-zero, and its norm is bounded by $B'$. Finally, the encoded message $\mu_2\mu_1 + \mu_3(1-\mu_1)$ is precisely $\mu_1 ? \mu_2 : \mu_3$. □

As in [2, lemma 3.5], we can ensure that the noise of all the entries of a ciphertext have independent Gaussian or Sub-Gaussian distributions. Namely, we say that $f_{\mathbf{h}}^{-1}$ is $\beta$-subgaussian if for each $y \in H$, $f_h^{-1}(y)$ returns a short Sub-Gaussian vector of parameter $\beta \geq \eta_\varepsilon(\mathcal{L}_{\mathbf{h}})$. The noise propagation analysis of [2, lemma 3.5] can be extended as follows:

**Lemma 11 (All noises are Sub-Gaussian).** *Assume that $f_{\mathbf{h}}^{-1}$ is $\beta$-subgaussian for $\beta \geq \eta_\varepsilon(\mathcal{L}_{\mathbf{h}})$. In a circuit containing solely* GSW.Not, GSW.And *and* GSW.Mux *gates, and whose inputs are either fresh GLWE ciphertexts or the noiseless ciphertexts* 0 *and* $\mathbf{h}$*, the output ciphertext of each individual gate has*

*the form* $\mathbf{z} + \mu\mathbf{h}$ *where* $\mu$ *is the encoded bit and the* $\ell$*-coordinates of* $\mathtt{Phase}(\mathbf{z})$ *are statistically indistinguishable from independent Gaussian samples of* $\mathbb{T}_k$*. We define the noise parameter* $\sigma(\mathtt{Phase}(z))$ *as the maximum of these* $\ell$ *Gaussian parameters.*

Thus, we may work directly with the square subgaussian parameter of the noise, which follows pythagorean summation.

**Lemma 12 (Average noise of primitive gates).** *Assume that* $f_{\mathbf{h}}^{-1}()$ *is* $\sqrt{\beta}$*-subgaussian for some* $\beta > 0$*. Let* $\mathbf{c}_1 = \mathbf{z}_1 + \mu_1 \cdot \mathbf{h}$*,* $\mathbf{c}_2 = \mathbf{z}_2 + \mu_2 \cdot \mathbf{h}$*,* $\mathbf{c}_3 = \mathbf{z}_3 + \mu_3 \cdot \mathbf{h} \in H^\ell$ *be three ciphertexts of a circuit satisfying the constraints of Lemma 11, and whose Gaussian parameters satisfy* $\sigma(\mathtt{Phase}(\mathbf{z}_1))^2 \leq B$ *and* $\sigma(\mathtt{Phase}(\mathbf{z}_2))^2, \sigma(\mathtt{Phase}(\mathbf{z}_3))^2 < B'$ *for some* $B, B' \in \mathbb{R}^+$*. Then:*

$$\mathtt{GSW.Not}(\mathbf{c}_1) = \mathbf{z} + \mathtt{NOT}(\mu_1) \cdot \mathbf{h} \text{ with } \sigma(\mathtt{Phase}(\mathbf{z}))^2 = B \tag{7}$$

$$\mathtt{GSW.And}(\mathbf{c}_1, \mathbf{c}_2) = \mathbf{z}' + (\mu_1 \ \mathtt{AND} \ \mu_2) \cdot \mathbf{h} \text{ with } \sigma(\mathtt{Phase}(\mathbf{z}'))^2 \leq \ell\beta B + B' \tag{8}$$

$$\mathtt{GSW.Mux}(\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3) = \mathbf{z}'' + (\mu_1 ? \mu_2 : \mu_3) \cdot \mathbf{h} \text{ with } \sigma(\mathtt{Phase}(\mathbf{z}''))^2 \leq 2\ell\beta B + B' \tag{9}$$

Since (4), (5) and (6) define the same recurrence as (7), (8) and (9), we will express the end of the paper only in terms of lemma 10, but all the bounds we obtain on the $\|\mathtt{Noise}\|$ also apply to the $\sigma(\mathtt{Noise})^2$ under Lemma 12.

### 7.3 Homomorphically Evaluating Arbitrary Functions

The result of the following corollary was already obtained in [2]; it states that in a long chain of And gates where one of the bits is a fresh GLWE-GSW ciphertext, the noise increases in fact linearly instead of exponentially. Here, we invert the operands of the And gates, so that the overall noise in the resulting ciphertext is smaller if one associates long conjunctions on the right.

**Corollary 5 (Noise of Conjunctions).** *Suppose* $f_{\mathbf{h}}^{-1}()$ *is* $\beta$*-bounded for some* $\beta \in \mathbb{R}^+$*. Let* $\mathbf{c}_1, \ldots, \mathbf{c}_k \in H^\ell$ *be* $k$ *ciphertexts such that each* $\mathbf{c}_i = \mathbf{z}_i + \mu_i \cdot \mathbf{h}$ *where* $\|\mathtt{Phase}(\mathbf{z}_i)\|_\infty < B$ *for some* $B \in \mathbb{R}^+$*. Then:*

$$\mathtt{GSW.And}(\mathbf{c}_1, \mathtt{GSW.And}(\mathbf{c}_2, \ldots \mathtt{GSW.And}(\mathbf{c}_{k-1}, \mathbf{c}_k))) = \mathbf{z} + (\mu_1 \mu_2 \ldots \mu_k) \cdot \mathbf{h}$$

*where* $\|\mathtt{Phase}(\mathbf{z}\|_\infty \leq k\ell\beta B$*.*

*Proof.* Apply (5) by induction on $k$. □

Note that any boolean function with $k$ inputs can always be put into disjunctive normal form, *i.e.* a disjoint union of conjunctive terms. One way to homomorphically evaluate the result is to add the ciphertexts of all the terms, which indeed preserves the $\{0, 1\}$ message space. However, with this method, the resulting noise will be proportional to the number of terms in the disjunctive normal form, which may still be exponential in the number of inputs.

By using Mux-gates, we obtain the following corollary, which says that any function can be homomorphically evaluated in a trivial way, where the noise

grows proportionally to only the square root of the number of inputs. We recall that the truth table of a boolean function $\phi$ with $k$ variables is a vector $\mathcal{T}$ of length $2^k$ such that each $\mathcal{T}_j = \phi(e_0, \ldots, e_{k-1})$ where $j = \sum e_i 2^{k-1-i}$. The full binary decision diagram (BDD) of $\phi$ is a circuit representing a binary tree of Mux-gates, of depth $k$. The bottom level $k$ consists in $2^k$ leaves $X_{k,j}$, each one is set to $\mathcal{T}_j$. At each intermediate level $i$, we have $2^i$ nodes $X_{k,j} = \texttt{Mux}(\mu_i, X_{i+1,2j+1}, X_{i+1,2j})$. By definition, the root $X_{0,0}$ thus contains $\phi(\mu_0, \ldots, \mu_{k-1})$. See Fig. 1 for an example of truth table and its associated BDD circuit.

**Corollary 6 (Evaluating arbitrary functions).** *Assume that $f_{\mathbf{h}}^{-1}()$ is $\beta$-bounded for some $\beta \in \mathbb{R}^+$. Let $\phi$ be any boolean function with $k$ inputs, and let $\mathbf{c}_1, \ldots, \mathbf{c}_k \in H^\ell$ be $k$ ciphertexts such that each $\mathbf{c}_i = \mathbf{z}_i + \mu_i \cdot \mathbf{h}$ where $\sigma(\mathbf{z}_i)^2 < B$ for some $B \in \mathbb{R}^+$. Then, the Mux-based Binary Decision Diagram of $\phi$ computes a ciphertext $\mathbf{c} = \mathbf{z} + \phi(\mu_1, \ldots, \mu_k).\mathbf{h}$ where $\|\mathbf{z}\|_\infty \leq 2k\ell\beta B$.*

*Proof.* To evaluate the full BDD of $\phi$ homomorphically, we just replace each leaf $X_{k,j}$ by noiseless ciphertexts $\mathcal{T}_j.\mathbf{h}$, each bit $\mu_i$ by their encryption $\mathbf{c_i}$, and each Mux gate by `GSW.Mux`. Apply (6) by induction on the depth, then all nodes $X_{i,j}$ at depth $i$ have a noise bounded by $2(k-i)\beta B$. □

In the previous corollary, the full BDD tree of the function $\phi$ contains a number of nodes which is exponential in the number of inputs. If the output noise is indeed really small, the time complexity to evaluate all the gates remains large when the simulated function has many variables. For some useful functions, like the bootstrapping function in the next section, many of the subtrees turn out to be equal. By merging them, the complexity to evaluate the circuit can be significantly reduced.

**Corollary 7 (Faster Evaluation of arbitrary functions).** *Assume that $f_{\mathbf{h}}^{-1}()$ is $\beta$-bounded for some $\beta \in \mathbb{R}^+$. Let $\phi$ be any boolean function with $k$ inputs, and let $\mathbf{c}_1, \ldots, \mathbf{c}_k \in H^\ell$ be $k$ ciphertexts such that each $\mathbf{c}_i = \mathbf{z}_i + \mu_i \cdot \mathbf{h}$ where $\|\texttt{Phase}(\mathbf{z}_i)\|_\infty < B$ for some $B \in \mathbb{R}^+$. We call $\mathcal{N}(\phi)$ the number of disctinct subtrees in the full Binary Decision Diagram of $\phi$. Then we can compute a ciphertext $\mathbf{c} = \mathbf{z} + \phi(\mu_1, \ldots, \mu_k).\mathbf{h}$ where $\|\texttt{Phase}(\mathbf{z})\|_\infty \leq 2k\ell\beta B$ by evaluating $\mathcal{N}(\phi)$ homomorphic `GSW.Mux`-gates.*

*Proof.* It suffices to evaluate the ciphertext value in the root of the $\mathcal{N}(\phi)$ subtrees by increasing depth. There are at most two different leaves, whose ciphertext values 0 and $\mathbf{h}$ are given. Whenever we need to evaluate a subtree of non zero depth $i$, the left and right subtrees have by definition already been fully evaluated, since their depth $i-1$ is strictly smaller. The root of the current tree is the `GSW.Mux` of $\mathbf{c}_i$ and the two subtrees roots. The last ciphertext to be evaluated is the root of the full tree, which contains the encrypted result. □

In the above corollary, Nerode's partitioning algorithm for reducing deterministic automata can efficiently list the $\mathcal{N}(\phi)$ identical subtrees. Indeed, a binary decision diagram is just the mirror graph of a deterministic accessible automata.

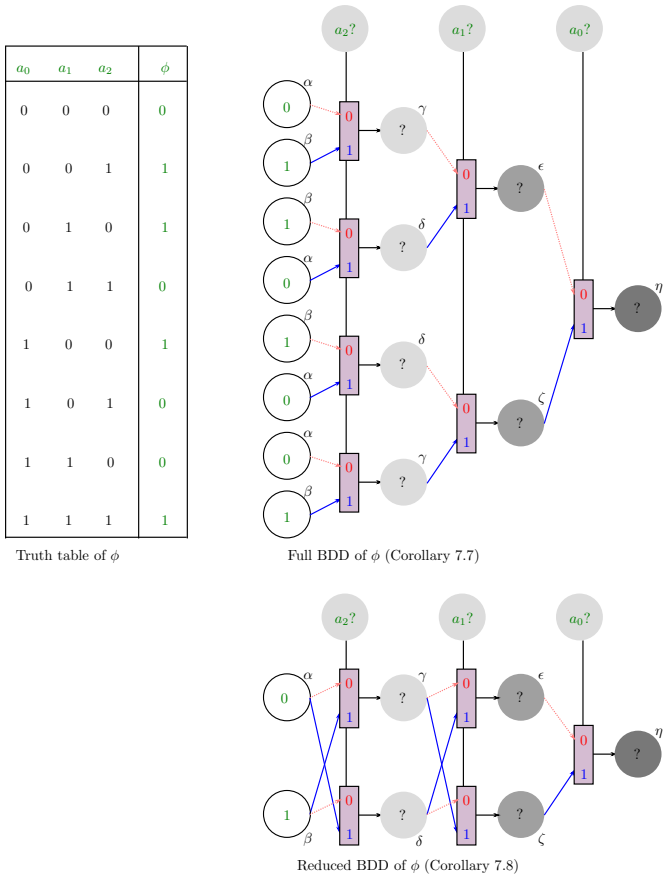


**Fig. 1.** Homomorphic evaluation of an arbitrary boolean function

Any truth table (top left) can be batch-converted to a full BDD (Binary decision diagram) circuit (top right). When evaluated homomorphically, the output noise is proportional to the square root of the number of inputs. After merging identical subtrees, which can be efficiently achieved using Nerode's partitioning algorithm, one obtains the reduced BDD circuit (bottom right), which may sometimes be much smaller. Here, the function $\phi$ was the Xor of all inputs. In this case, there are only two different subtrees per levels, and the reduced BDD is therefore polynomial in the number of inputs.

More generally, the GSW.Mux gate allows to homomorphically evaluate the transitions of a deterministic automata, which leads to the following lemma.

**Lemma 13 (Recognizing arbitrary rational langages).** *Let $\mathcal{L}$ be an arbitrary rational language of $\{0,1\}^*$ and $\mathcal{N}(\tilde{\mathcal{L}})$ be the number of residuals of the mirror language of $\mathcal{L}$. Given $k$ ciphertexts $\mathbf{c}_1, \ldots, \mathbf{c}_k$ of a message $\mathbf{w} = w_1, \ldots, w_k$, one can compute a ciphertext $\mathbf{c} = \mathbf{z} + \mathcal{L}(\mathbf{w}).\mathbf{h}$ where $\mathcal{L}(\mathbf{w}) = 1$ iff $\mathbf{w} \in \mathcal{L}$ and $\|\mathtt{Phase}(\mathbf{z})\|_\infty \leq 2k\beta B$ by evaluating $k\mathcal{N}(\tilde{\mathcal{L}})$ GSW.Mux-gates.*

*Proof.* Let $\mathcal{A} = (Q, i, T_0, T_1, F)$ be a minimal deterministic automata of the mirror language $\tilde{\mathcal{L}}$ where $Q$ is the set of states, $i \in Q$ is the initial state, $T_0, T_1$ are the two transitions functions from $Q$ to $Q$ and $F$ is the set of final states. Note that $\#Q = \mathcal{N}(\tilde{\mathcal{L}})$. We initialize $\#Q$ noiseless ciphertexts $X_{q,0}$ for $q \in Q$ with $X_{q,0} = \mathbf{h}$ if $q \in F$ and $X_{q,0} = 0$ otherwise. Then for each letter we compute the transition as follow: $X_{q,j} = \mathtt{GSW.Mux}(c_j, X_{T_1(q),j-1}, X_{T_0(q),j-1})$. And we output $X_{i,k}$. We write $\mathbf{a} \equiv \mathbf{b}$ when two ciphertexts $\mathbf{a}$ and $\mathbf{b} \in H^\ell$ encrypt the same bit. Then we have $X_{i,k} \equiv X_{T_{w_k}(i),k-1} \equiv \ldots \equiv X_{T_{w_1}(T_{w_2}\ldots(T_{w_k}(i))\ldots),0}$, which encrypts 1 iff $T_{w_1}(T_{w_2}\ldots(T_{w_k}(i))\ldots) \in F$, *i.e.* iff $w_k \ldots w_1$ is accepted by $\mathcal{A}$ iff $w_1 \ldots w_k \in \mathcal{L}$. This proves correctness.

For the complexity, each $X_{q,j}$ is computed with a single GSW.Mux gate and the noise increases as in the previous corollary since the fresh-GSW.Mux depth of the circuit is $k$. □

Many arithmetic functions, including addition, multiplication and comparison correspond to polynomial-size deterministic automata, and in the next section, we prove that a direct application of Corollary 7 suffices to bootstrap the whole system, turning it into a fully homomorphic one.

### 7.4 Simple Bootstrapping Circuit with Polynomial Noise

Bootstrapping refers to Gentry's homomorphic decryption, which allows to turn suitable somewhat-homomorphic schemes into fully-homomorphic schemes. Here, the decryption procedure is simply the GLWE decryption of the first entry.

The GLWE decryption of $(d, c) \in G \times \mathbb{T}$ consists in computing $c - \hat{s}(d) \in \mathbb{T}$ and deciding whether it is closer to $\frac{1}{2}$ or 0. If the secret $\hat{s}$ has $n-1$ bits $(s_1, \ldots, s_{n-1})$, this sum can be linearized as $c - \sum_{i=1}^{n-1} s_i d_i$ where $c, d_1, \ldots, d_{n-1} \in \mathbb{T}$ are publicly computable. Necessarily $n$ is always $\leq \ell$. Furthermore, if the noise of $(d, c)$ is $\frac{1}{8}$-bounded, these $n$ values can be rounded to their nearest multiple of $\frac{1}{4n}$ without affecting the result of the decryption. Thus, bootstrapping a ciphertext $(d, c) \in H$ is equivalent to homomorphically evaluate on $(s_1, \ldots, s_{n-1})$, its *bootstrapping boolean function* $\phi(x_1, \ldots, x_{n-1})$ which returns the most significant bit of $c' - \sum_{i=1}^{n-1} s_i d'_i$ where $c' = \lfloor 4n(c + \frac{1}{4}) \rceil, d'_i = \lfloor d_i \rceil$ are known integers modulo $4n$.

**Lemma 14 (Simple Bootstrapping).** *Given a GSW ciphertext $\mathbf{c} = \mathbf{z} + \mu\mathbf{h} \in H^\ell$, s.t. $\|\mathtt{Phase}(\mathbf{z})\|_\infty < \frac{1}{8}$, whose first entry is $(d, c) \in H$, its bootstrapping function $\phi$ satisfies $\mathcal{N}(\phi) \leq 4n^2$. Therefore, if $f_{\mathbf{h}}^{-1}$ is 1-bounded, given*

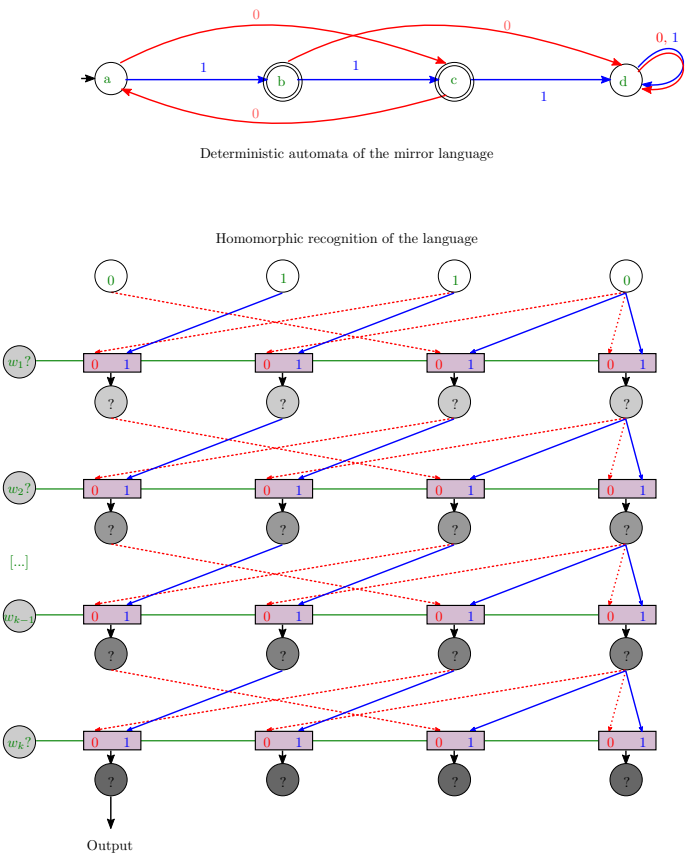


**Fig. 2.** A Mux-based circuit recognizing a regular language $\mathcal{L}$ and the corresponding deterministic automata of the mirror $\tilde{\mathcal{L}}$.

The automata on the top recognizes the regular expression $\tilde{\mathcal{L}} = (110|00)^*(1|11|0)$. The circuit below recognizes its mirror $\mathcal{L} = ((110|00)^*(11|0))|((101)^*1)$. The circuit transcripts the mirror graph of the automata: final states $b, c$ are initialized with a noiseless ciphertext of 1, non-final states $a, d$ with 0. All transitions are reversed and point to a `GSW.Mux` controlled by the encrypted current letter. After processing all letters, the final output is the ciphertext corresponding to the automata's initial state $a$. As explained in the proof of Lemma 13, The result encrypts one iff the word is in $\mathcal{L}$.

*the bootstrapping key* $(BK_i)_{i\in[1,n-1]}$ *where* $BK_i$ *encrypts the* $i$*-th bit of* $\hat{s}$ *with* $\|\texttt{Noise}(BK_i)\|_\infty \leq B$*, one can compute a ciphertext* $\mathbf{c}' = \mathbf{z}' + \mu\mathbf{h}$ *of the same message where* $\|\texttt{Phase}(\mathbf{z}')\|_\infty < 2n\ell B$ *by evaluating at most* $4n^2$ `GSW.Mux` *gates.*

*Proof.* The expression of $\phi$ as a sum proves that for all $(x_1, \ldots, x_k)$ and $(y_1, \ldots, y_k)$ and $(z_{k+1}, \ldots, z_n)$ such that $\sum_{i=1}^{k} x_i d'i = \sum_{i=1}^{k} y_i d'i \mod 4n$, then $\phi(x_1, \ldots, x_k, z_{k+1}, \ldots, z_n) = \phi(y_1, \ldots, y_k, z_{k+1}, \ldots, z_n)$. This proves that for each index $k \in [0, n-1]$, there are at most $4n$ distinct partial functions of $\phi$ by fixing the first $k$ coordinates. And thus, $\mathcal{N}(\phi) \leq 4n^2$. The rest follows from Corollary 7. □

Recall that under the hypothesis of lemma 11, the max-norm of the noise can be replaced by its square Gaussian parameter. It follows that the GLWE-GSW scheme is fully homomorphic according to Gentry's blueprint by design, as soon as the initial GLWE Gaussian parameter is $1/\tilde{O}(\ell^{1.5})$, which represents a time *vs* noise trade-off compared to the [2] proposal, and shows that the construction of a homomorphic circuit amounts to analyzing a few intrinsic parameters of the computed function.

We can obtain a [2]-like variant of the decryption circuit with $\tilde{O}(n)$ gates, and with noise overhead $\tilde{O}(n)$ by composing homomorphic functions, as in the following lemma.

**Lemma 15 (CRT variant).** *Given a GLWE ciphertext* $c \in H$*, the gadget* $\mathbf{h}$ *and its 1-bounded function* $f_{\mathbf{h}}^{-1}$*, let* $q = \prod_{i=1}^{t} p_i$ *be an integer larger than* $4n$ *where* $p_i$ *are* $t = O(\log(n))$ *distinct primes where* $p_i = O(\log(n))$*. We suppose that the encryption of each individual bit* $BK_i$ *of* $\hat{s}$ *are provided as bootstrapping key with* $\|\texttt{Noise}(BK_i)\|_\infty \leq B$*. Then given as input a ciphertext of a bit* $\mu$*, one can compute a ciphertext* $\mathbf{c} = \mathbf{z} + \mu\mathbf{h}$ *of the same bit with noise* $\|\texttt{Phase}(\mathbf{z})\|_\infty = \tilde{O}(\ell^3)$ *by evaluating* $\tilde{O}(\ell)$ *homomorphic Mux-gates.*

*Proof.* It suffices to evaluate $\phi'(y_1, \ldots, y_t)$ where each $y_j = f_j(s_1, \ldots, s_{n-1})$ for the following functions:

- $f_j$ for $j \in [1, t]$, takes $n-1$ bits and returns the $O(\log(p_j))$ bits of $c' - \sum s_i d'_i$ modulo $p_j$. ($f_j$ can be viewed as $O(\log(p_j))$ boolean functions with a single bit output).
- $\phi'$ takes $t$ numbers modulo $p_1, \ldots, p_t$, and hence $O(\log(n)\log\log(n))$ input bits, and returns the most significant bit of their CRT lift modulo $q$.

As before, the expression of each $f_j$ as a sum proves that $\mathcal{N}(f_j) \leq (n-1)p_j$ and that $\mathcal{N}(\phi') \leq q.\sum_{k=1}^{t} \log(p_k)$. By Lemma 7, the homomorphic ciphertext of each bit of $y_j$ has noise norm $\tilde{O}(\ell n)$. Thus the output noise norm of $y$ is $\tilde{O}(\ell^2 n)$. The total number of `GSW.Mux` gates is $\sum_{j=1}^{t} \log_2(p_j)\mathcal{N}(f_j) + \mathcal{N}(\phi') = \tilde{O}(n)$ □

Interestingly, the noise overhead obtained from this lemma is smaller than the one from [2]. We compare our FHE scheme to previous ones in Table 1. In that table, the GLWE group is taken as $\mathbb{Z}_q^n$, which makes our scheme based on the standard LWE assumption. In this case, we could take $\ell = O(n \log q)$.

| Schemes | Primitive Gates | #Gates in Boots. | Boots. noise overhead |
|---|---|---|---|
| BGV12 [10] | And, Xor, Const. | $\tilde{O}(n^2)$ | $n^{O(\log n)}$ |
| Bra12 [9] | And, Xor, Const. | $\tilde{O}(n^2)$ | $n^{O(\log n)}$ |
| GSW13 [21] | And, Xor, Nand, Const. | $\tilde{O}(n^2)$ | $n^{O(\log n)}$ |
| BV14 [12] | And, Xor, Const. | $\tilde{O}(n^{6/\varepsilon})$ | $\tilde{O}(n^{\varepsilon})$ |
| AP14 [2] | And, Not, Const. | $\tilde{O}(n)$ | $\tilde{O}(n^2)$ |
| DM15 [15] | Nand, Const. | $\tilde{O}(n)$ | $\tilde{O}(n^{1.5})$ |
| Ours | Mux, Not, Const. | $\tilde{O}(n^2)$ | $\tilde{O}(n)$ |
| Ours (with CRT) | Mux, Not, Const. | $\tilde{O}(n)$ | $\tilde{O}(n^{1.5})$ |

**Table 1.** Comparisons of LWE-based FHE Schemes

This table compares the primitive gates, the number of homomorphic gates to bootstrap, and the average bootstrapping noise overhead, *i.e.* the ratio between the noise parameter of the refreshed ciphertext and the (fresh) noise of the bootstrapping key. Multiplying this value by $O(\sqrt{n})$ gives the minimal underlying GLWE inverse error rate to make the scheme fully homomorphic. And multiplying this value by an additional $O(n)$ gives the SIVP approx. factor using the quantum worst-case to average-case reduction. Const. means constant gates (*i.e.* noiseless ciphertexts) Finally, note that the construction in [15] necessarily relies on algebraic lattices, and that this scheme does not support somewhat homomorphic evaluation of expressions: it must be bootstrapped between each individual NAND gate.

### Acknowledgements

Part of this work has been supported by Fonds Unique Interministériel (FUI) through the CRYPTOCOMP project and the EIT Digital project HC@WORKS, China's 973 Program (Grant 2013CB834205), and NSFC's Key Project (Grant 61133013).

## References

1. M. Ajtai. Generating hard instances of lattice problems. In *STOC*, pages 99–108, 1996.
2. J. Alperin-Sheriff and C. Peikert. Faster bootstrapping with polynomial error. In *Crypto*, pages 297–314, 2014.
3. B. Applebaum, D. Cash, C. Peikert, and A. Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In *Crypto*, LNCS 5677, pages 595–618, 2009.
4. G. Baumslag, N. Fazio, A. Nicolosi, V. Shpilrain, and W. E. Skeith. Generalized learning problems and applications to non-commutative cryptography. In *Proc. ProvSec '11*, volume 6980 of *LNCS*, pages 324–339. Springer, 2011.
5. A. Becker, N. Gama, and A. Joux. A sieve algorithm based on overlattices. *LMS J. Comput. Math.*, 17(A):49–70, 2014. Cryptology ePrint Archive, report 2013/685.
6. D. Bleichenbacher. On the generation of DSA one-time keys. Draft of September 13, 2004. Short presentation at the rump session of CRYPTO 2005.
7. A. Blum, A. Kalai, and H. Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. *J. ACM*, 50(4):506–519, 2003.

8. D. Boneh and R. Venkatesan. Hardness of computing the most significant bits of secret keys in Diffie-Hellman and related schemes. In *Crypto*, volume 1109 of *LNCS*, 1996.
9. Z. Brakerski. Fully homomorphic encryption without modulus switching from classical gapsvp. In *Crypto*, pages 868–886, 2012.
10. Z. Brakerski, C. Gentry, and V. Vaikuntanathan. (leveled) fully homomorphic encryption without bootstrapping. In *ITCS*, pages 309–325, 2012.
11. Z. Brakerski, A. Langlois, C. Peikert, O. Regev, and D.Stehlé. Classical hardness of learning with errors. In *Proc. of 45th STOC*, pages 575–584. ACM, 2013.
12. Z. Brakerski and V. Vaikuntanathan. Lattice-based FHE as secure as PKE. In *ITCS*, pages 1–12, 2014.
13. J.-Y. Cai. *Algorithmic Number Theory: 4th International Symposium, ANTS-IV Leiden, The Netherlands, July 2-7, 2000. Proceedings*, chapter The Complexity of Some Lattice Problems, pages 1–32. Springer Berlin Heidelberg, Berlin, Heidelberg, 2000.
14. Y. Chen and P. Q. Nguyen. BKZ 2.0: Better lattice security estimates. In *Proc. of Asiacrypt*, pages 1–20, 2011.
15. L. Ducas and D. Micciancio. FHEW: Bootstrapping homomorphic encryption in less than a second. In *Eurocrypt*, pages 617–640, 2015.
16. N. Fazio, K. Iga, A. R. Nicolosi, L. Perret, and W. E. Skeith III. Hardness of learning problems over burnside groups of exponent 3. *Designs, Codes and Cryptography*, pages 1–12, 2013.
17. N. Gama, M. Izabachène, P. Q. Nguyen, and X. Xie. Structural lattice reduction: Generalized worst-case to average-case reductions and homomorphic cryptosystems. *To appear soon on IACR Cryptology ePrint Archive*, 2016, 2016.
18. N. Gama and P. Q. Nguyen. Predicting Lattice Reduction. In *Eurocrypt*, 2008.
19. C. Gentry and S. Halevi. Implementing Gentry's fully-homomorphic encryption scheme. In *Advances in Cryptology - Proc. EUROCRYPT 2011*, volume 6632 of *Lecture Notes in Computer Science*, pages 129–148. Springer, 2011.
20. C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *STOC*, 2008.
21. C. Gentry, A. Sahai, and B. Waters. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In *Crypto*, pages 75–92, 2013.
22. D. Goldstein and A. Mayer. On the equidistribution of Hecke points. *Forum Math.*, 15(2):165–189, 2003.
23. A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Ann.*, 261:513–534, 1982.
24. A. Lubotzky. The expected number of random elements to generate a finite group. *J. Algebra*, 257(2):452–459, 2002.
25. D. Micciancio. Almost perfect lattices, the covering radius problem, and applications to Ajtai's connection factor. *SIAM J. Comput.*, 34(1):118–169, 2004.
26. D. Micciancio and C. Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In *Eurocrypt '12*, LNCS. Springer-Verlag, 2012.
27. D. Micciancio and C. Peikert. Hardness of SIS and LWE with small parameters. In *Crypto '13*, volume 8042 of *LNCS*, pages 21–39, 2013.
28. D. Micciancio and O. Regev. Worst-case to average-case reductions based on Gaussian measures. In *SIAM J. Comput.*, 2007.
29. L. J. Mordell. On some arithmetical results in the geometry of numbers. *Compositio Mathematica*, 1:248–253, 1935.

30. P. Q. Nguyen and I. E. Shparlinski. The insecurity of the digital signature algorithm with partially known nonces. *J. Cryptology*, 15(3):151–176, 2002.
31. P. Q. Nguyen and I. E. Shparlinski. Counting co-cyclic lattices. *CoRR*, abs/1505.06429, 2015. Preprint.
32. I. Pak. On probability of generating a finite group. Preprint, 1999.
33. A. Paz and C.-P. Schnorr. Approximating integer lattices by lattices with cyclic factor groups. In *Proc. of ICALP*, pages 386–393, 1987.
34. C. Peikert. Public-key cryptosystems from the worst-case shortest vector problem. In *STOC*, pages 333–342. ACM, 2009.
35. C. Peikert. An efficient and parallel gaussian sampler for lattices. In *Crypto*, LNCS 6223, pages 80–97. Spinger-Verlag, 2010.
36. O. Regev. Lattices in computer science #12: Average-case hardness. Regev's webpage, 2004.
37. O. Regev. On lattices, learning with errors, random linear codes, and cryptography. In *STOC*, pages 84–93, 2005.
38. M. van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan. Fully homomorphic encryption over the integers. In *Eurocrypt*, pages 24–43, 2010.

# A Missing Algorithms

All missing algorithms were sketched in the main body. Here, we provide explicit descriptions of these algorithms.

## A.1 Alg. 5: Sampling lattices of given factor group

**Algorithm 5** Sampling lattices of given factor group

**Input:** Integer $m \geq 1$ and a finite abelian group $G = \mathbb{Z}_{q_1} \times \cdots \times \mathbb{Z}_{q_k}$ such that $1 \leq k \leq m$.
**Output:** A random lattice from the uniform distribution over $\mathcal{L}_{G,m}$.
1: Generate elements $g_1, \ldots, g_m$ uniformly at random from $G$ until the $g_i$'s generate $G$.
2: Return the lattice $\mathcal{L}_{\mathbf{g}}$ where $\mathbf{g} = (g_1, \ldots, g_m) \in G^m$.

## A.2 Alg. 6: Unbalanced Reduction

## A.3 Alg. 7: Bootstrapping

**Algorithm 6** Unbalanced Reduction

**Input:** an $n \times m$ basis $B$ of an integer lattice $L \subseteq \mathbb{Z}^m$ and a target length $\sigma \in \mathbb{Q}^+$. More generally, $B$ can be any $n$-dimensional projected block $B = B'_{[i,i+n-1]}$ of some basis $B'$ of $L \subseteq \mathbb{Z}^m$.

**Output:** an $n \times n$ unimodular matrix $U$ such that $C = UB$ satisfies $\|c_i^*\| \leq \sigma$ for $i \geq 2$ and $\|\mathbf{c}_1\| \leq n\sigma\delta_\sigma(B)$.

$C \leftarrow B$, $U \leftarrow I_n$ and compute the Gram-Schmidt matrices $\mu$ and $C^*$
If $\|\mathbf{c}_i^*\| \leq \sigma$ for all $i$, **return** $U$
**for** $i = k-1$ downto 1 where $k$ is the largest index such that $\|\mathbf{c}_k^*\| > \sigma$ **do**
    **if** $\|\mathbf{c}_i^*\| \leq \sigma$ **then**
        $\alpha \leftarrow \lfloor -\mu_{i+1,i} \rceil$
    **else**
        $\alpha \leftarrow \left\lceil -\mu_{i+1,i} + \frac{\|\mathbf{c}_{i+1}^*\|}{\|\mathbf{c}_i^*\|}\sqrt{(\|\mathbf{c}_i^*\|/\sigma)^2 - 1} \right\rceil$
    **end if**
    $(\mathbf{c}_i, \mathbf{c}_{i+1}) \leftarrow (\mathbf{c}_{i+1} + \alpha \cdot \mathbf{c}_i,\ \mathbf{c}_i)$, $(\mathbf{u}_i, \mathbf{u}_{i+1}) \leftarrow (\mathbf{u}_{i+1} + \alpha \cdot \mathbf{u}_i,\ \mathbf{u}_i)$ and update the GS matrices $\mu$ and $C^*$.
**end for**
**return** $U$

**Algorithm 7** Bootstrapping algorithm

**Input:** A GLWE ciphertext $c \in H$, the gadget $\mathbf{h}$ and its functions $f_{\mathbf{h}}^{-1}$, and the bootstrapping key $(BK_{i,j})_{i\in[1,\ell],j\in[1,n]}$ where $BK_{i,1}, \ldots, BK_{i,n}$ are encryptions of the $n = \log_2(\ell) + 3$ most significant bits of $\texttt{Phase}(h_i)$.

**Output:** A GLWE-GSW ciphertext $\mathbf{c}' \in H^\ell$ encoding the same bit as $c$ with polynomial noise.

$\mathbf{x} \leftarrow f_{\mathbf{h}}^{-1}(c) \in \{0,1\}^\ell$
$p \leftarrow 0$
Set the initial state $(X_{0,0}, ..., X_{0,8\ell-1})$ where $X_{i,j} = 1$ iff $j \in [2\ell, 6\ell]$
**for** each $i \in [1, \ell]$ s.t. $x_i = 1$ **do**
    **for** $j = 1$ to $n$ **do** ▷ This loop adds $\texttt{Phase}(h_i)$ to the state
        $p \leftarrow p + 1$
        **for** $k = 0$ to $8\ell - 1$ **do** ▷ This loop adds $2^{n-j}$ to the state iff $BK_{i,j} = 1$
            $X_{p,k} \leftarrow \texttt{GSW.Mux}(BK_{i,j}, X_{p-1,k-2^{n-j} \bmod 8\ell}, X_{p-1,k})$
        **end for**
    **end for**
**end for**
**return** $\mathbf{c}' = X_{p,0}$ ▷ This is the final rounding.