Lattice Trapdoors and IBE from Middle-Product LWE

Alex Lombardi[1], Vinod Vaikuntanathan[1], and Thuy Duong Vuong[2]

[1] MIT, Cambridge MA 02139, USA
[2] Stanford University, Stanford, CA 94305, USA **

Abstract. Middle-product learning with errors (MP-LWE) was recently introduced by Rosca, Sakzad, Steinfeld and Stehlé (CRYPTO 2017) as a way to combine the efficiency of Ring-LWE with the more robust security guarantees of plain LWE. While Ring-LWE is at the heart of *efficient* lattice-based cryptosystems, it involves the choice of an underlying ring which is essentially arbitrary. In other words, the effect of this choice on the security of Ring-LWE is poorly understood. On the other hand, Rosca et al. showed that a new LWE variant, called MP-LWE, is as secure as Polynomial-LWE (another variant of Ring-LWE) *over any of a broad class of number fields*. They also demonstrated the usefulness of MP-LWE by constructing an MP-LWE based public-key encryption scheme whose efficiency is comparable to Ring-LWE based public-key encryption. In this work, we take this line of research further by showing how to construct Identity-Based Encryption (IBE) schemes that are secure under a variant of the MP-LWE assumption. Our IBE schemes match the efficiency of Ring-LWE based IBE, including a scheme in the random oracle model with keys and ciphertexts of size $\tilde{O}(n)$ (for n-bit identities).

We construct our IBE scheme following the lattice trapdoors paradigm of [Gentry, Peikert, and Vaikuntanathan, STOC'08]; our main technical contributions are introducing a new leftover hash lemma and instantiating a new variant of lattice trapdoors compatible with MP-LWE.

This work demonstrates that the efficiency/security tradeoff gains of MP-LWE can be extended beyond public-key encryption to more complex lattice-based primitives.

Keywords: Middle-Product LWE · Identity-Based Encryption · Lattice Trapdoors.

1 Introduction

Cryptographic schemes based on the polynomial learning with errors problem (PLWE) [23] and the ring learning with errors problem (RLWE) [13] have the advantage of having key size and algorithm runtime that are quasi-linear in the security parameter. However, their security guarantees are not as strong as that of the original learning with errors problem (LWE) [20].

One of the main differences between these two settings is that the PLWE problem parametrized by some (say, irreducible) polynomial f, denoted PLWE$^{(f)}$, is only known to be as hard as a worst-case problem on some class of lattices *that depends*

** Work done while at MIT

on the polynomial f, which could possibly be easier to solve for some choices of f as compared to others. In particular, we do not have a clear understanding of the relative hardness of $\mathrm{PLWE}^{(f)}$ for different f, making it hard for a cryptosystem designer to pick the right f. In contrast, with the LWE problem, there is no such ambiguity. For essentially any choice of possible modulus q, LWE is as hard as worst-case problems on *arbitrary* lattices [20, 17]. In summary, the concrete efficiency gains of RLWE and PLWE have only been obtained through a trade-off involving making both quantitatively and qualitatively more questionable security assumptions.

Recently, following on an earlier work of Lyubashevsky [11] who initiated the study of ring-independent assumptions, Rosca et al. [21] introduced the "middle-product learning with errors" assumption (MP-LWE), a new variant of LWE that uses the "middle product" of polynomials modulo q. For any f in a broad class of polynomials, they show a reduction from $\mathrm{PLWE}^{(f)}$ to the MP-LWE problem, which is defined independently of any such f, freeing the cryptosystem designer from making an essentially arbitrary choice of f. They also describe a public key encryption (PKE) scheme that has quasi-linear (optimal) key size and algorithm runtime, while being IND-CPA secure under the MP-LWE assumption. Thus, they obtain a public-key encryption scheme with the same efficiency gains over LWE-based PKE as enjoyed by PLWE-based schemes, but prove security under a worst-case assumption on a comparatively broader class of lattices.

While the idea of using MP-LWE as an alternative to Ring-LWE, as proposed by [21], is intriguing, it is only currently known how to construct plain public-key encryption from MP-LWE. In this work, we consider and make progress on the following question.

Can we instantiate more complex lattice-based primitives using middle-product LWE while maintaining the improved efficiency/security tradeoff?

Indeed, it is explicitly left open by [21] to instantiate more complex lattice-based primitives, such as lattice trapdoors [8] and their applications, using MP-LWE.

1.1 Our Results

We construct an Identity-Based Encryption (IBE) scheme based on MP-LWE. This scheme is IND-CPA secure in the random oracle model under the MP-LWE assumption and has quasi-linear key size and algorithm runtime.

Our construction follows the "lattice trapdoors" paradigm of [8]. Specifically, we construct a "dual" of the public key encryption scheme in [21], then combine the dual scheme with Micciancio-Peikert style lattice trapdoors [15] to obtain the IBE scheme. In addition to our IBE scheme in the random oracle model, we sketch how techniques for constructing IBE schemes in the standard model [3, 2, 7] can also be adapted to the MP-LWE setting using our lattice trapdoors.

Our main IBE construction from MP-LWE can be stated informally as follows.

Theorem 1 (Informal). *For any* $\epsilon \geq 2^{-\mathsf{poly}(\log n)}$*, there is a* (T, ϵ)*-secure IBE scheme (in the random oracle model) under the* (T', ϵ') *MP-LWE assumption with* $T' \approx T$*,* $\epsilon' \approx \epsilon$*. This scheme has quasi-linear* $\tilde{O}(n)$ *key size and encryption runtime.*

By (T, ϵ)-security, we mean that any T-time adversary fails to break the primitive/assumption with advantage greater than ϵ. In particular, assuming that MP-LWE is hard for $T(n) = 2^{\alpha n}$-time adversaries, we show that our IBE scheme is hard to break in time roughly T with better than some inverse quasi-polynomial advantage.

Our IBE scheme demonstrates that the better efficiency/security trade-off obtained by [21] for public key encryption can be extended to more expressive cryptographic primitives such as IBE. Tables 1 and 2 compare the efficiency of our PKE and IBE schemes to prior works.

Table 1. Summary of parameters of our "dual Regev"-like public encryption scheme from MP-LWE versus prior ones.

PKE scheme	LWE based [19]	RLWE based [13]	MP-LWE based ("primal"-[21], "dual"-this work)
pk size	$\tilde{O}(n^2)$	$\tilde{O}(n)$	$\tilde{O}(n)$
sk size	$\tilde{O}(n)$	$\tilde{O}(n)$	$\tilde{O}(n)$
Enc/Dec runtime per encrypted bit	$\tilde{O}(n)$-amortized	$\tilde{O}(1)$	$\tilde{O}(1)$

IBE scheme	LWE based [8, 15]	RLWE based [15]	MP-LWE based (this work)
mpk size	$\tilde{O}(n^2)$	$\tilde{O}(n)$	$\tilde{O}(n)$
msk size	$\tilde{O}(n^2)$	$\tilde{O}(n)$	$\tilde{O}(n)$
Enc/Dec runtime per encrypted bit	$\tilde{O}(n)$-amortized	$\tilde{O}(1)$	$\tilde{O}(1)$

Table 2. Summary of parameters of our identity-based encryption (IBE) scheme from MP-LWE versus prior ones that are from LWE and Ring-LWE.

1.2 Technical Overview

As mentioned before, we follow the “lattice trapdoors” paradigm of [8]. We first recall the approach of [8] for constructing IBE from LWE. The high-level idea is as follows: using a random oracle H, design a key pair (distribution) $(\mathsf{mpk}, \mathsf{msk})$ such that for any identity id, $\mathsf{pk}_{\mathsf{id}} := (\mathsf{mpk}, H(\mathsf{id}))$ is a valid public key for some public key encryption scheme PKE. In order for this to yield an IBE scheme, it must be possible to derive a corresponding secret key $\mathsf{sk}_{\mathsf{id}}$, using msk, from the public value $H(\mathsf{id})$. This is achieved in the following way.

- **Step 1: Dual Regev Encryption.** First, [8] constructs a “dual” variant of Regev encryption [20] in which public keys are (statistically close to) uniformly random. In slightly more detail, public keys have the form $(A, u = Ar)$ for $A \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$ and

$r \xleftarrow{\$} \chi^m$ for some distribution χ of "small numbers." This step is done so that in an associated IBE scheme, $u = H(\mathsf{id})$ can be interpreted as (part of) a dual Regev public key.
- **Step 2: Lattice Trapdoors.** The most technically complicated step in [8] is designing an alternative procedure $\mathsf{TrapGen}$ that outputs a (statistically close to) uniformly random matrix A along with a *trapdoor* T_A that allows for sampling, given an input $u \in \mathbb{Z}_q^n$, a random preimage $r \leftarrow \chi^m \mid Ar = u$. This step allows for efficient secret key extraction from a public key (A, u).
- **Step 3: Constructing IBE.** As has been implicitly described already, [8] then write down an IBE scheme with a master key pair (A, T_A) sampled using $\mathsf{TrapGen}$, so that encryption for an identity id uses dual Regev encryption with public key $(A, u = H(\mathsf{id}))$, and secret keys $\mathsf{sk}_{\mathsf{id}} = r$ can be extracted from $(A, u = H(\mathsf{id}))$ using $\mathsf{msk} = T_A$.

We now describe how we instantiate this framework using middle-product LWE.

Step 1: MP-LWE based dual Regev. Our first step is to develop an analogue of dual Regev encryption based on middle-product LWE. We first recall from [21] that the middle-product learning with errors assumption over $\mathbb{Z}_q$ with degrees (n, d) is that the distribution

$$\{(a_i, a_i \odot_d s + e_i)_{i=1}^t\}$$

is computationally pseudorandom, where s is a uniformly random degree n polynomial,[3] each a_i is a uniformly random degree $n - d$ polynomial, each e_i is a random "small" degree d polynomial, and $a_i \odot_d s$ is the "middle product" consisting of the d "middle terms" of the polynomial product $a \cdot s$. [21] show that this assumption suffices to construct a "primal Regev" public key encryption scheme, and show that this assumption follows from the hardness of $\mathsf{PLWE}^{(f)}$ for various polynomials f.

We would like to develop a "dual Regev" public-key encryption scheme similar to the PKE of [21], and a natural approach suggests itself (based on [21]): Let $a_1, \ldots, a_t$ be t i.i.d. degree n polynomials, let $r_1, \ldots, r_t$ be t i.i.d. degree k polynomials (for some additional parameter k), and set

$$\left(\mathsf{pk} = (a_1, \ldots, a_t, u = \sum a_i r_i), \mathsf{sk} = (r_1, \ldots, r_t)\right).$$

Encrypting a message μ then consists of sampling a random MP-LWE secret s and outputting middle products[4] $(a_i \odot s, +2e_i)_{i \le t}$ along with $u \odot s + 2e' + \mu$. We would then like to argue, as in [21], that the security of this scheme follows from MP-LWE (with secret s).

Technical Challenges. However, there are two main issues that arise from this approach to Step 1 (which both arise again when implementing Step 2):

[3] The parameters are slightly simplified for exposition

[4] We omit some details regarding degrees; it turns out that the middle products $a_i \odot s$ will have a different degree from the middle product $u \odot s$ in order to get decryption correctness.

- We need a new variant of the leftover hash lemma to argue that the polynomial $u = \sum a_i r_i$ is (statistically close to) uniformly random. The reason that previous leftover hash lemma variants seem insufficient is related to the fact that the map $r \mapsto \sum_i a_i r_i$ has a larger range (degree $n + k$ polynomials) than its domain (degree k polynomials); this stands in contrast to the PLWE setting, where $r \mapsto \sum a_i r_i \pmod{f}$ is reduced modulo a degree n polynomial f. Indeed, the hash function $h_{a_1,\ldots,a_t}(r_1, \ldots, r_t) = \sum a_i r_i$ is *not* 2-universal, unlike the hash function considered in [21], so we have to argue the desired statistical indistinguishability directly. We state and prove our new variant of the LHL in Section 4. We use techniques from [14] designed to prove a variant of the LHL in the *Ring-LWE* setting; however, these techniques must be substantially modified to handle the distinction between multiplication of bounded-degree polynomials in $\mathbb{Z}_q[x]$ (as in our setting) and multiplication over rings of the form $R_q = \mathbb{Z}_q[x]/f(x)$ (as in the RLWE/PLWE setting).
- Middle-product LWE as defined in [21] does not seem directly applicable to (the security of) our dual-Regev encryption scheme; the reason is that the "coefficient polynomials" $(a_1, \ldots, a_t, u)$ do not all have the same degree.[5] In order to prove security, we have to consider a new variant of MP-LWE in which "coefficient polynomials" $\{a_i\}$ can have different degrees; we consider a variant in which the adversary can specify a new degree d_i for each sample in advance. In Section 3, we show that (a simple modification of) the [21] reduction from PLWE to MP-LWE carries over to our variant of MP-LWE, which we call "degree-parametrized MP-LWE."

After addressing these two difficulties, the approach outlined in Step 1 can be made to work, yielding a dual-Regev encryption scheme based on MP-LWE.

Step 2: Lattice Trapdoors for MP-LWE. Having developed a variant of dual Regev encryption, we next turn to constructing lattice trapdoors [8] that are compatible with this new encryption scheme. To do so, we make use of the work [15], which gives a highly general roadmap for constructing lattice trapdoors.

Following the basic idea of [15], our procedure TrapGen will produce polynomials $a_1, \ldots, a_t, a_{t+1}, \ldots, a_{t'}$ such that for $t < i \le t'$,

$$a_i = c_i - \sum_{j=1}^{t} a_j w_{ij}$$

for random, small "trapdoor" polynomials $\{w_{ij}\}$ and specific polynomials $\{c_j = 2^u x^{dv}\}$ which are our analogue to the "G matrix" in plain LWE-based constructions. Similarly to the Ring-LWE setting, we can think of these polynomials as "structured matrices" by associating a polynomial $g(x)$ with the "multiplication by $g(x)$" matrix acting on a vector space of bounded-degree polynomials. We show that with this choice of polynomials $\{c_j\}$, the matrix A corresponding to $(a_1, \ldots, a_{t'})$ has a "G-trapdoor" (as defined

[5] In fact, after introducing lattice trapdoors, our scheme will be modified so that *three* different degrees will be used rather than two (as it is currently written).

in [15]) that can be efficiently described using our trapdoor $\{w_{ij}\}$. The preimage sampling algorithm of [15] can then be adapted to yield a corresponding preimage sampling algorithm for polynomial sum-products; moreover, we show that our preimage sampling algorithm has the same $\widetilde{O}(n)$ efficiency gain over plain LWE that is enjoyed by Ring-LWE based constructions. See Section 5 for more details.

Finally, we note that we are implicitly relying on resolutions to both "technical challenges" mentioned above in this step; our leftover hash lemma is what guarantees that $\mathsf{TrapGen}$ outputs a distribution $\{(a_1, \ldots, a_{t'})\}$ that is statistically close to uniform, while our "degree-parametrized MP-LWE" allows us to redesign our dual Regev scheme to have public key $(a_1, \ldots, a_{t'}, u)$ such that $\{a_1, \ldots, a_t\}$ and $\{a_{t+1}, \ldots, a_{t'}\}$ have different degrees.

Step 3: Constructing IBE Schemes. Given our variant of dual Regev encryption from MP-LWE (Step 1) and our variant of lattice trapdoors compatible with this new encryption scheme (Step 2), constructing IBE is fairly straightforward given prior work. We describe constructions analogous to those of [8] (in the random oracle model) and [3] (in the standard model) using our new tools.

Remark On Concrete Efficiency/Security. As usual, some care is required when comparing the efficiency of various LWE-based cryptosystems to take into account their expected levels of security. We give an overview of the comparison of LWE/RLWE/MPLWE-based IBE schemes using concrete security [5].

The concrete security of all relevant lattice-based cryptosystems is based on assumptions of the following form.

Definition 1 ((T, ϵ)-secure X-LWE, Informal). *Any time T adversary breaks the X-LWE assumption with advantage at most ϵ.*

Our main IBE construction from MP-LWE (as stated in Theorem 1) constructs (T, ϵ)-secure IBE from roughly (T, ϵ)-secure LWE, *as long as* $\epsilon \geq 2^{-\mathsf{poly}\log n}$. This technical limitation is due to the achievable parameters of our leftover hash lemma (which was already implicitly noted in [21]) when used in a standard hybrid argument to prove security of the IBE scheme. This barrier also appears in the Ring-LWE context (see, e.g., the signature scheme of [15]) when quasi-linear efficiency is desired. However, these works (and ours) still attain a meaningful form of concrete security because security is proved against adversaries that run in exponential time (assuming that the LWE variants are exponentially secure).

In addition, with some more work, it is possible to improve Theorem 1 to hold for smaller values of ϵ (without sacrificing efficiency). This improved security proof is based on the use of Renyi divergence (as opposed to statistical distance), as demonstrated in [4], and will appear in the full version of this paper.

1.3 Organization

The rest of the paper is organized as follows. In Section 2, we review basic definitions and other preliminaries. In Section 3, we introduce and prove the hardness of our "degree-parametrized MP-LWE," a slight variant on the original defintion. In Section 4, we prove our new leftover hash lemma for bounded degree polynomials. In Section 5, we use our new LHL in combination with [15] to develop lattice trapdoors for middle-product LWE. Finally, in Section 6, we combine our new tools to construct MP-LWE based dual Regev public-key encryption and IBE.

2 Preliminaries

Negligible Functions. We use n to denote the security parameter. We use standard big-O notation to classify the growth of functions, and say that $f(n) = \tilde{O}(g(n))$ if $f(n) = O(g(n) \cdot \log^c n)$ for some fixed constant c. We let $\mathrm{poly}(n)$ denote an unspecified function $f(n) = O(n^c)$ for some constant c. We say that a function $f(n)$ is negligible (denoted $f(n) = \mathsf{negl}(n)$) if $f(n) = o(n^{-c})$ for every fixed constant c. We say that a probability (or fraction) is overwhelming if it is $1 - \mathsf{negl}(n)$.

Statistical and Computational Indistinguishability. The statistical distance between two distributions X and Y over a countable domain Ω is defined to be $\Delta(X, Y) := \frac{1}{2} \cdot \sum_{d \in \Omega} |X(d) - Y(d)|$. We say that two distributions X, Y (formally, two ensembles of distributions indexed by n) are statistically indistinguishable if $\Delta(X, Y) = \mathsf{negl}(n)$, and write $X \approx_s Y$.

Two ensembles of distributions $\{X_n\}$ and $\{Y_n\}$ are computationally indistinguishable if for every probabilistic poly-time machine A, $|\Pr[A(1^n, X_n) = 1] - \Pr[A(1^n, Y_n) = 1]| = \mathsf{negl}(n)$; we denote this relationship by $X \approx_c Y$.

Polynomials. Let R be a ring. For any integer $d > 0$ and any set $S \subseteq R$, we let $S^{<d}[x]$ denote the set of polynomials in $R[x]$ of degree $< d$ whose coefficients are in S. For any distribution χ defined over R, let $\chi^d[x]$ denote the distribution on polynomials in $R^{<d}[x]$ where each coefficient is sampled independently according to χ.

Given a polynomial $a = \sum_{i=0}^{d-1} a_i x^i \in R^{<d}[x]$, define the coefficient vector of a as $\mathbf{a} := (a_0, \cdots, a_{d-1})^T \in R^d$. In particular, for any $0 \leq i \leq d-1$, $\mathbf{a}_i$ denotes the coefficient of x^i in a.

Probability. For any distribution X defined on a countable domain Ω, we define the *collision probability*

$$\mathrm{CP}(X) := \Pr_{X, X' \text{ i.i.d.}}[X = X']$$

as well as the *Renyi entropy* of X,

$$H_2(X) := \log_2 \frac{1}{\mathrm{CP}(X)},$$

and the *min-entropy* of X,

$$H_\infty(X) := \log_2 \min_{x \in \Omega} \frac{1}{\Pr[X = x]}.$$

We remark that $H_2(X) \geq H_\infty(X)$ for all distributions X. For a finite set Ω, we let $U(\Omega)$ denote the uniform distribution over Ω, and we use the notation $X \overset{\$}{\leftarrow} \Omega$ to denote that X is sampled uniformly at random from Ω. For a distribution χ over $\mathbb{R}$, let χ^k denote the distribution over $\mathbb{R}^k$ where each coordinate is independently sampled from χ. For a distribution D over $\mathbb{R}^k$, let $D[x]$ be the distribution over $\mathbb{R}^{<k}[x]$ where the coefficient vector of polynomials is sampled from D.

2.1 Identity-Based Encryption

We recall the standard syntax and definition of security under chosen-plaintext and chosen-identity attack [6, 1] for IBE. An IBE scheme consists of four algorithms.

- A setup algorithm $\mathsf{IBESetup}$ (on input 1^n) outputs a master public key mpk and master secret key msk.
- A secret key extraction algorithm $\mathsf{IBEExtract}$, given msk and an identity id, outputs a secret key $\mathsf{sk}_{\mathsf{id}}$.
- An encryption algorithm Enc, given the master public key mpk, an identity id, and a message m, outputs a ciphertext c.
- A decryption algorithm Dec, given the secret key sk and a ciphertext c, outputs a message m.

We require that an IBE scheme $\mathsf{IBE} = (\mathsf{IBESetup}, \mathsf{IBEExtract}, \mathsf{Enc}, \mathsf{Dec})$ satisfies two properties.

- **Correctness:** For all identities id and messages m, we have

  $$\Pr[\mathsf{Dec}(\mathsf{sk}_{\mathsf{id}}, \mathsf{Enc}(\mathsf{mpk}, \mathsf{id}, m)) = m] = 1 - \mathsf{negl}(n),$$

  where the probability is taken with respect to the randomness of $\mathsf{IBESetup}$, $\mathsf{IBEExtract}$, Enc, and Dec.
- **Security:** Security is defined by the following game (defined for a given PPT adversary $\mathcal{A}$).
  - $(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{IBESetup}(1^n)$ is sampled. Define a (randomized) oracle $\mathcal{O}(\cdot)$ that on input id outputs $\mathsf{IBEExtract}(\mathsf{msk}, \mathsf{id})$.
  - $\mathcal{A}^{\mathcal{O}(\cdot)}(\mathsf{mpk})$ outputs a challenge $(\mathsf{id}^*, m_0, m_1)$.
  - $b \leftarrow U(\{0,1\})$ is sampled uniformly at random.
  - $\mathsf{ct}^* \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{id}^*, m_b)$ is sampled.
  - $\mathcal{A}^{\mathcal{O}(\cdot)}(\mathsf{ct}^*)$ outputs a bit b' and wins if (1) $\mathcal{O}(\mathsf{id}^*)$ was not queried and (2) $b' = b$.

  We say that the scheme is secure if every PPT adversary $\mathcal{A}$ wins the above game with probability at most $\frac{1}{2} + \mathsf{negl}(n)$

2.2 Middle Product of Polynomials [21]

Definition 2 ([21], Definition 3.1). *Let d_a, d_b, d, k be integers such that $d_a + d_b - 1 = d + 2k$. The middle product $\odot_d : R^{<d_a}[x] \times R^{<d_b}[x] \to R^{<d}[x]$ is defined to be the map*

$$(a, b) \mapsto a \odot_d b = \lfloor \frac{(a \cdot b) \mod x^{k+d}}{x^k} \rfloor = \sum_{k \leq i+j \leq k+d-1} (\boldsymbol{a}_i \boldsymbol{b}_j) x^{i+j},$$

*where **a** and **b** are the coefficient vectors of a and b, respectively. In other words, $a \odot_d b$ is obtained by deleting the k highest and k lowest degree terms of the polynomial product $a \cdot b$, then dividing the remaining d terms by x^k.*

Immediately from Definition 2, the middle product is commutative, i.e., $a \odot_d b = b \odot_d a$ for all polynomials a, b. The middle product also satisfies a "quasi-associative" property.

Lemma 1 ([21]). *Let $d, k, n > 0$. For all $r \in R^{<k+1}[x], a \in R^{<n}[x], s \in R^{<n+d+k-1}[x]$, we have*

$$r \odot_d (a \odot_{d+k} s) = (r \cdot a) \odot_d s.$$

2.3 Lattices

An n-dimensional lattice Λ is a discrete additive subgroup of $\mathbb{R}^n$. A lattice has rank $k \leq m$ if it is generated as the set of all $\mathbb{Z}$-linear combinations of some k linearly independent *basis* vectors $\mathbf{B} = (\mathbf{b}_1, \cdots, \mathbf{b}_k)$; we say Λ is full-rank if $k = m$. The dual lattice Λ^* is the set of all $v \in \text{Span}_{\mathbb{R}}(\Lambda)$ such that $\langle v, x \rangle \in \mathbb{Z}$ for every $x \in \Lambda$. If $\mathbf{B}$ is a basis of Λ, then $\mathbf{B}^* = B(B^t B)^{-1}$ is a basis of Λ^*. Note that when Λ is full-rank, $\mathbf{B}$ is invertible and hence $\mathbf{B}^* = \mathbf{B}^{-1}$.

For any set $\mathbf{S} = (s_1, \cdots, s_k)$ of linearly independent vectors, let $\tilde{\mathbf{S}}$ denote its Gram-Schmidt orthogonalization, defined iteratively in the following way: $\tilde{s}_1 = s_1$, and for each $i = 2, \cdots, n$, $\tilde{s}_i$ is the component of s_i orthogonal to span$(s_1, \cdots, s_{i-1})$.

For positive integers n, q and any matrix $A \in \mathbb{Z}_q^{n \times m}$, let $\Lambda^{\perp}(A) := \{z \in \mathbb{Z}^m : Az = 0 \mod q\}$. For $u \in \mathbb{Z}_q^n$ such that $\exists t \in \mathbb{Z}_q^m$ satisfying $At = u$, let $\Lambda_u^{\perp}(A) := \{z \in \mathbb{Z}^m : Az = u \mod q\} = \Lambda^{\perp}(A) + t$.

Gaussian Distributions

Definition 3 (Continuous Gaussian distribution). *For a positive semidefinite matrix $\Sigma \in \mathbb{R}^{n \times n}$, the continuous Gaussian distribution D_Σ is the probability distribution over $\mathbb{R}^n$ whose density is proportional to $\rho_\Sigma(x) = \exp(-\pi x^T \Sigma^{-1} x)$.*

Definition 4 (Discrete Gaussian distribution). *Given countable set $S \subset \mathbb{R}^n$ and $s > 0$, the discrete Gaussian distribution $D_{S,\sigma,\mathbf{c}}$ is the probability distribution over S whose density is proportional to $\rho_{\sigma,\mathbf{c}}(x) := \exp(-\pi \cdot ||\boldsymbol{x} - \mathbf{c}||^2/\sigma^2)$. That is, for $x \in S$: $D_{S,\sigma,\mathbf{c}} = \frac{\rho_{\sigma,\mathbf{c}}(x)}{\rho_{\sigma,\mathbf{c}}(S)}$. If $\mathbf{c} = 0$, we can omit $\mathbf{c}$ and write $D_{S,\sigma}$ instead.*

As usual, we will make use of various statistical properties of the discrete Gaussian $D_{\Lambda,\sigma}$ when σ is large compared to the *smoothing parameter* of the lattice Λ, defined below.

Definition 5 ([16]). *For any n-dimensional lattice Λ and real $\epsilon > 0$, the smoothing parameter $\eta_\epsilon(\Lambda)$ is defined to be the smallest real $s > 0$ such that $\rho_{1/s}(\Lambda^* \setminus \{\boldsymbol{0}\}) \leq \epsilon$.*

The following lemma gives an upper bound on the smoothing parameter of Λ in terms of its Gram-Schmidt basis $\tilde{B}$.

Lemma 2 ([8], Theorem 3.1). *Let $\Lambda \subset \mathbb{R}^n$ be a lattice with basis $\mathbf{B}$ and real $\epsilon > 0$. Then,*

$$\eta_\epsilon(\Lambda) \leq ||\tilde{\mathbf{B}}|| \cdot \sqrt{\ln(2n(1+\epsilon^{-1}))/\pi}.$$

where $\tilde{\mathbf{B}} = (\tilde{b}_1, \cdots, \tilde{b}_k)$ is the Gram-Schmidt orthogonalization of $\mathbf{B}$ as defined in Section 2.3, and $||\tilde{\mathbf{B}}|| = \max_{i\in[k]} ||\tilde{b}_i||$.

We will make use of tail bounds on $D_{\Lambda,\sigma}$ (for σ larger than the smoothing parameter).

Lemma 3 ([8], Lemma 2.9). *For any ϵ > 0, any $\sigma \geq \eta_\epsilon(\mathbb{Z})$, and any $t > 0$, we have*

$$\Pr_{x \leftarrow D_{\mathbb{Z},\sigma,c}} [|x - c| \geq t \cdot \sigma] \leq 2e^{-\pi t^2} \cdot \frac{1+\epsilon}{1-\epsilon}.$$

In particular, for $\epsilon \in (0, 1/2)$ and $t \geq \omega(\sqrt{\log n})$, the probability that $|x - c| \geq t \cdot \sigma$ is negligible in n.

In addition, we will make use of *entropy* bounds on $D_{\Lambda,\sigma}$ (again for σ sufficiently large). In order to prove these bounds, we first recall the following approximation.

Lemma 4 ([18], Lemma 2.10). *Let $\Lambda \subset \mathbb{R}^d$ be a full-rank lattice. For any $s \geq \eta_\epsilon(\Lambda)$, we have*

$$s^d \det(\Lambda^*) \cdot (1-\epsilon) \leq \rho_s(\Lambda) \leq s^d \det(\Lambda^*) \cdot (1+\epsilon).$$

Using Lemma 4, we can bound $H_\infty(D_{\Lambda,\sigma})$ and $H_2(D_{\Lambda,\sigma})$.

Lemma 5. *For a full-rank lattice $\Lambda \subset \mathbb{R}^d$ and discrete Gaussian distribution $\chi = D_{\Lambda,\sigma}$ with parameters $\epsilon \in (0,1), \delta \in (0,1)$, and $\sigma \geq \max(\sqrt{2}, \delta^{-1}) \cdot \eta_\epsilon(\Lambda)$, we have*

$$2^{-H_\infty(\chi)} \leq \delta^d \frac{1+\epsilon}{1-\epsilon}$$

and

$$\mathrm{CP}(\chi) \leq \left(\frac{\delta}{\sqrt{2}}\right)^d (\frac{1+\epsilon}{1-\epsilon})^2.$$

Proof. Using Lemma 4, we obtain the bound

$$D_{\Lambda,\sigma}(\mathbf{x}) \leq \frac{1}{\sigma^d \det(\Lambda^*) \cdot (1-\epsilon)}$$

for all $\mathbf{x} \in \Lambda$. Moreover, we assumed that $\sigma\delta \geq \eta_\epsilon(\Lambda)$, so by Lemma 4 we also have

$$1 \leq \rho_{\sigma\delta}(\Lambda) \leq (\sigma\delta)^d \det(\Lambda^*) \cdot (1+\epsilon).$$

Combining this with the first inequality, we see that

$$D_{\Lambda,\sigma}(\mathbf{x}) \leq \delta^d \frac{1+\epsilon}{1-\epsilon},$$

yielding the desired bound on $2^{-H_\infty(\chi)}$. In order to bound $\mathrm{CP}(\chi)$, we write

$$\mathrm{CP}(\chi) = \sum_{x \in \Lambda} D_{\Lambda,\sigma}(x)^2 = \rho_\sigma(\Lambda)^{-2} \sum_{x \in \Lambda} \rho_\sigma(x)^2 = \rho_\sigma(\Lambda)^{-2} \cdot \rho_{\sigma/\sqrt{2}}(\Lambda),$$

where the last equality uses the identity $\rho_\sigma(x)^2 = \rho_{\sigma/\sqrt{2}}(x)$. Since we assumed that $\sigma > \delta\sigma \geq \eta_\epsilon(\Lambda)$, Lemma 4 (applied three times, to parameters σ, $\frac{\sigma}{\sqrt{2}}$ and $\sigma\delta$) tells us that

$$\begin{aligned} \rho_\sigma(\Lambda)^{-2}\rho_{\sigma/\sqrt{2}}(\Lambda) &\leq \frac{(\sigma/\sqrt{2})^d \det(\Lambda^*)(1+\epsilon)}{\sigma^{2d}\det^2(\Lambda^*)(1-\epsilon)^2} \\ &= \left(\frac{\delta}{\sqrt{2}}\right)^d \left(\frac{1+\epsilon}{1-\epsilon}\right)^2 \frac{1}{(\sigma\delta)^d \det(\Lambda^*)(1+\epsilon)} \\ &\leq \left(\frac{\delta}{\sqrt{2}}\right)^d (\frac{1+\epsilon}{1-\epsilon})^2, \end{aligned}$$

completing the proof.

2.4 Polynomials and Matrices

For a vector $\mathbf{v} \in \mathbb{R}^n$, let $||v||, ||v||_\infty$ denote the Euclidean and sup norm respectively. We define the largest singular value of a matrix $A \in \mathbb{R}^{m\times n}$ as $\sigma_1(A) := \max_{||u||=1} ||Au||$.

Lemma 6. *For any matrix $A \in \mathbb{R}^{m\times n}$, we have $\sigma_1(A) \leq \sqrt{mn}\max_{i,j}|A_{ij}|$.*

We will make use of the following matrix representation of polynomial multiplication.

Definition 6. *Let R be a ring and $d, k, > 0$ be positive integers. For any polynomial $a \in R^{<k}[x]$ of degree less than k, let $T^{k,d}(a)$ denote the matrix in $R^{(k+d-1)\times d}$ whose i-th column, for $i = 1, \cdots, d$, is given by the coefficients of $x^{i-1} \cdot a$, listed from lowest to highest degree. In particular, $T^{k,1}(a)$ is the coefficient vector **a** of the polynomial a (possibly with zeros appended).*

Lemma 7. *For $\ell, k, d > 0, a \in R^{<k}[x], b \in R^{<\ell}[x], T^{k,\ell+d-1}(a)\cdot T^{\ell,d}(b) = T^{\ell+k-1,d}(a\cdot b)$.*

Definition 7 ([21], from [12]). *Let $f \in \mathbb{Z}[x]$ have degree m. The expansion factor of f is defined as*

$$\mathrm{EF}(f) := \max_{g \in \mathbb{Z}^{<2m-1}[x]} \frac{||g \mod f||_\infty}{||g||_\infty}.$$

For our purposes, we are interested in polynomials with poly(n)-bounded expansion factors. One such class [12] is the family of all $f = x^m + h$ where $\deg(h) \leq m/2$ and $||h||_\infty \leq \mathrm{poly}(n)$.

Definition 8. *Let f be a monic polynomial over a ring R of degree m. Define the (Hankel) matrix $\boldsymbol{M}_f \in R^{m \times m}$ such that for $1 \leq i, j \leq m$, $(\boldsymbol{M}_f)_{i,j}$ is the constant coefficient of $x^{i+j-2} \mod f$.*

Under suitable conditions on f, the matrix $\mathbf{M}_f$ is guaranteed to be invertible.

Lemma 8. *If $f \in R[x]$ has constant coefficient f_0 which is invertible in R, then $\boldsymbol{M}_f$ is an invertible matrix.*

Proof. Rearranging the columns of $\mathbf{M}_f$ gives a triangular matrix whose diagonal is the constant coefficient of f.

Moreover, when $f \in \mathbb{Z}[x]$, we will make use of singular value bounds on $\mathbf{M}_f$ and related matrices in terms of the expansion factor of f. For a matrix $A \in \mathbb{R}^{m \times n}$ let $A^{(d)}$ denote the matrix whose rows are the first d rows of A.

Lemma 9. *For any $f \in \mathbb{Z}[x]$, $\sigma_1(\boldsymbol{M}_f^{(d)}) \leq \sqrt{d}\,\mathrm{EF}(f)$.*

Remark 1. This inequality generalizes and improves on [Theorem 2.8]MPLWE by a factor of $\sqrt{d}$.

Proof. We want to show that for all nonzero vectors $u \in \mathbb{R}^m$, the following inequality holds:

$$\frac{||\mathbf{M}_f^{(d)}u||}{||u||} \leq \sqrt{d}\,\mathrm{EF}(f).$$

We first note that because $\mathbb{Q}$ is dense in $\mathbb{R}$, it suffices to show the same inequality for all nonzero $u \in \mathbb{Q}^n$. Moreover, since the inequality is scale-invariant, we may further reduce to the case where $u \in \mathbb{Z}^m$.

Given any nonzero vector $u \in \mathbb{Z}^m$, we define $v := \mathbf{M}_f u$. Then, letting $g \in \mathbb{R}^{<m}[x]$ denote the degree $< m$ polynomial with coefficient vector u, we know by [Lemma 2.4]MPLWE that v_i is the constant coefficient of $x^{i-1} \cdot g \mod f$. Thus,

$$|v_i| \leq ||g \cdot x^{i-1} \mod f||_\infty \leq \mathrm{EF}(f)||x^{i-1} \cdot g||_\infty = \mathrm{EF}(f)||u||_\infty.$$

We conclude that

$$\frac{||\mathbf{M}_f^{(d)}u||}{||u||} \leq \sqrt{d}\frac{||u||_\infty}{||u||}\,\mathrm{EF}(f) \leq \sqrt{d}\,\mathrm{EF}(f),$$

where the last inequality holds because $||u||_\infty \leq ||u||$.

3 Degree-Parametrized MP-LWE

In this section, we define and consider a variant of MP-LWE in which samples generated from a fixed secret s (which are polynomials with coefficients in $\mathbb{Z}_q$) can have varying (pre-specified) degrees. This is in contrast to the variant considered in [21], in which all samples have the same degree. We then prove a hardness reduction relating polynomial LWE (PLWE) to our variant of MP-LWE, which we call *degree-parametrized* MP-LWE.

For the rest of the paper, we will let $\mathbb{R}_q$ denote $\mathbb{R}/q\mathbb{Z}$.

Definition 9 (Degree-Parametrized MP-LWE). *Let $n > 0, q \geq 2, m > 0, \mathbf{d} \in [\frac{n}{2}]^t$, and let χ be a distribution over $\mathbb{R}_q$. For $s \in {\mathbb{Z}_q}^{<n-1}[x]$, we define the distribution $\mathrm{MP}_{q,n,\boldsymbol{d},\chi}(s)$ over $\prod_{i=1}^{t}({\mathbb{Z}_q}^{<n-\boldsymbol{d}_i}[x] \times {\mathbb{R}_q}^{<\boldsymbol{d}_i}[x])$ as follows.*

- *For each $i \in [t]$, sample $a_i \xleftarrow{\$} {\mathbb{Z}_q}^{<n-\boldsymbol{d}_i}[x])$ and sample $e_i \leftarrow \chi^{\boldsymbol{d}_i}$ (interpreted as a degree $< \boldsymbol{d}_i$ polynomial).*
- *Output $(a_i, b_i := a_i \odot_{\boldsymbol{d}_i} s + e_i)_{i \in [t]}$.*

The (degree-parametrized) MP-LWE problem consists of distinguishing between arbitrarily many samples from $\mathrm{MP}_{q,n,\boldsymbol{d},\chi}(s)$ and the same number of samples from $\prod_{i=1}^{t} U({\mathbb{Z}_q}^{<n-\boldsymbol{d}_i}[x] \times {\mathbb{R}_q}^{<\boldsymbol{d}_i}[x])$ with non-negligible probability over the choice of $s \xleftarrow{\$} {\mathbb{Z}_q}^{<n-1}[x]$.

Following [21], we show that degree-parametrized MP-LWE is as hard as the polynomial-LWE problem PLWE_f for a wide class of polynomials f. The reduction is effectively the same as that of [21], although we obtain better parameters due to an improved singular value bound on the matrix $\mathbf{M}_f$ (Lemma 6). We recall the definition of PLWE_f, taken from [21].

Definition 10 (PLWE). *Let $q \geq 2, m > 0$, f a polynomial of degree m, χ a distribution over $\mathbb{R}[x]/f$. The decision problem $\mathsf{PLWE}^{(f)}_{q,\chi}$ consists in distinguishing between arbitrarily many samples*

$$\{a \xleftarrow{\$} \mathbb{Z}_q[x]/f, e \leftarrow \chi : (a, a \cdot s + e)\}$$

and the same number of samples from $U(\mathbb{Z}_q[x]/f \times \mathbb{R}_q[x]/f)$ with non-negligible probability over choice of $s \xleftarrow{\$} \mathbb{Z}_q[x]/f$.

Theorem 2 (Hardness of MP-LWE). *Let $n > 0, q \geq 2, t > 0, \boldsymbol{d} \in [\frac{n}{2}]^t$, and $\alpha \in (0,1)$. For $S > 0$, let $\mathcal{F}(S, \boldsymbol{d}, n)$ be the set of polynomials in $\mathbb{Z}[x]$ that are monic, have constant coefficient coprime with q, have degree m in $\bigcap_{i=1}^{t}[\boldsymbol{d}_i, n - \boldsymbol{d}_i]$ and satisfy $\mathrm{EF}(f) < S$. Then, there exists a ppt reduction from $\mathsf{PLWE}^{(f)}_{D_{\alpha \cdot q}}$ for any $f \in \mathcal{F}(S, \boldsymbol{d}, n)$ to $\mathsf{MPLWE}_{q,n,\boldsymbol{d},D_{\alpha' \cdot q}}$ with $\alpha' = \alpha \cdot \sqrt{\frac{n}{2}} \cdot S$.*

Proof. For $d \in [n/2]$ and any polynomial f of degree $m \in [d, n-d]$, we describe a ppt mapping

$$\phi_{n,d} : (a, b) \in \mathbb{Z}_q[x]/f \times \mathbb{R}_q[x]/f \mapsto (a', b') \in {\mathbb{Z}_q}^{<n-d}[x] \times {\mathbb{R}_q}^{<d}[x].$$

We will then show that ϕ maps $U(\mathbb{Z}_q[x]/f \times \mathbb{Z}_q[x]/f)$ to $U({\mathbb{Z}_q}^{<n-d}[x] \times {\mathbb{Z}_q}^{<d}[x])$ and maps a random PLWE sample (with secret s) to a random MP-LWE sample with secret s' depending on s. This mapping (with slightly different parameters) was previously defined in [21].

Let $(a, b) \in \mathbb{Z}_q[x]/f \times \mathbb{R}_q[x]/f$ be an input pair. Then, the pair $(a', b') \leftarrow \phi_{n,d}(a, b)$ is sampled by the following process.

- Define the matrix $\Sigma = (\alpha' q)^2 \mathbf{I}_d - (\alpha q)^2 \mathbf{J}_d \cdot \mathbf{M}_f^{(d)}$, where $\mathbf{I}_d$ denotes the $d \times d$ identity matrix and $\mathbf{J}_d$ denotes the $d \times d$ anti-diagonal matrix.

- Sample $h \xleftarrow{\$} \mathbb{Z}_q{}^{<n-d-m}[x]$ and $\epsilon \leftarrow D_\Sigma$.
- Set $a' = a + h \cdot f$ and set b' to be the polynomial with coefficient vector $\mathbf{b}' = \mathbf{J}_d \cdot \mathbf{M}_f^{(d)} \cdot \mathbf{b} + \epsilon$.

We note that the matrix Σ above is positive definite (and hence the distribution D_Σ is well-defined) by the following calculation: using Lemma 9,

$$\sigma_1\left((\alpha q)^2 \mathbf{J}_d \cdot \mathbf{M}_f^{(d)}\right) \leq \alpha q \cdot \sigma_1(\mathbf{J}_d) \cdot \sigma_1(\mathbf{M}_f^{(d)}) \leq \alpha q \cdot 1 \cdot \sqrt{d}\,\mathrm{EF}(f) < \alpha' \cdot q.$$

We first show that if $(a, b) \xleftarrow{\$} \mathbb{Z}_q[x]/f \times \mathbb{R}_q[x]/f$, then (a', b') is distributed uniformly on the set $\mathbb{Z}_q{}^{<n-d}[x] \times \mathbb{R}_q{}^{<d}[x]$. Since a and h are uniformly random distributed in $\mathbb{Z}_q{}^{<m}[x]$ and $\mathbb{Z}_q{}^{<n-d-m}[x]$ respectively, we see that a' is uniformly distributed over $\mathbb{Z}_q{}^{<n-d}[x]$. Moreover, if b is uniformly distributed in $\mathbb{R}_q[x]/f$, then its coefficient vector $\mathbf{b}$ is uniformly distributed in $\mathbb{R}_q^m$. Since $\mathbf{J}$ and $\mathbf{M}_f$ are invertible (see Lemma 8), $\mathbf{J}_d \cdot \mathbf{M}_f^{(d)} \cdot \mathbf{b}$ is therefore uniformly distributed (over $\mathbb{R}_q^d$), thus so is $\mathbf{b}'$ and its polynomial representation b'.

We next show that if (a, b) is a PLWE-sample, then (a', b') is a MP-LWE sample. Suppose that $b = a \cdot s + e$ for $s \in \mathbb{Z}_q[x]/f$ and error polynomial e with coefficient vector $\mathbf{e} \leftarrow \chi^d$. Let $s' \in \mathbb{Z}_q{}^{<n-1}[x]$ be defined so that it has coefficient vector

$$\mathbf{s}' = \mathbf{J}_{n-1} \cdot (\mathbf{B}_{n-1,f} \cdot \mathbf{M}_f \cdot \mathbf{s}),$$

where $\mathbf{B}_{n-1,f} \in \mathbb{Z}_q^{n-1 \times m}$ is defined so that the ith row of $\mathbf{B}_{n-1,f}$ is the coefficient vector of $x^{i-1} \mod f$. Moreover, define $e' \in \mathbb{R}_q{}^{<d}[x]$ to have coefficient vector

$$\mathbf{e}' = \mathbf{J}_d \cdot \mathbf{M}_f^{(d)} \cdot \mathbf{e} + \epsilon.$$

We refer to [21] for a proof that $b' = a' \odot_d s' + e'$. Since ϵ is sampled independent of $\mathbf{e}$, the distribution of $\mathbf{e}'$ is $D_{\alpha' \cdot q}$ by standard (continuous) Gaussian distribution identities.

As in [21], the collection of maps ϕ_{n,d_i} (ranging over all $i \in [t]$) can be used to implement a MP-LWE oracle using a PLWE oracle, and hence immediately give a reduction from $\mathsf{PLWE}^{(f)}_{D_{\alpha \cdot q}}$ for any $f \in \mathcal{F}(S, \mathbf{d}, n)$ to $\mathsf{MPLWE}_{q,n,\mathbf{d},D_{\alpha' \cdot q}}$.

4 A Leftover Hash Lemma for Polynomials

In this section, we state and prove Theorem 3, a Leftover Hash Lemma for polynomials with bounded degree. The closest previous work is [14], in which the author proves a Leftover Hash Lemma for elements of the ring $R := \mathbb{Z}_q[\alpha]/[\alpha^n - 1]$; the proof technique in [14] inspires our proof of Theorem 3. However, we encounter some subtleties as a result of working with bounded-degree polynomials; specifically, difficulties arise due to the fact that the set of bounded-degree polynomials is not closed under multiplication.

Let $q = \mathsf{poly}(n)$ be a sequence of prime numbers, so that $\mathbb{Z}_q$ is a field for every $q = q(n)$. For polynomials $z_1, \cdots, z_t \in \mathbb{Z}_q[x]$, we adopt the convention that $\gcd(z_1, \cdots, z_t)$ is always monic.

Our goal is to prove that the hash function

$$h_{a_1,\ldots,a_t}(z_1,\ldots,z_t) = \sum_{i=1}^{t} a_i z_i,$$

with hash key $\vec{a} = (a_1,\ldots,a_t)$ consisting of t polynomials drawn i.i.d. from $U(\mathbb{Z}_q{}^{<n}[x])$, extracts uniform randomness from high entropy sources of bounded-degree polynomials, in the special case of sources that are *product distributions.*

Following the approach of [14], we want to analyze, for any fixed input $\vec{z} = (z_1,\ldots,z_t)$, the distribution of outputs $H(\vec{z}) := h_{a_1,\ldots,a_t}(z_1,\ldots,z_t)$ over the choice of uniformly random hash key. In [14], the a_i and z_i are all elements of a ring R, so the set $\{h_{a_1,\cdots,a_t}(z_1,\cdots,z_t) | (a_1,\cdots,a_t) \in R^t\}$ is simply the ideal generated by $z_1,\cdots,z_t$. Moreover, a simple argument shows that $H(\vec{z})$ is uniform over this set. Here, however, we are working with bounded degree polynomials, so the characterization of $H(\vec{z})$ is not as immediate. Lemma 10 characterizes $H(\vec{z})$.

Lemma 10 (Range of hash output). *Consider $z_1,\cdots,z_t \in \mathbb{Z}_q{}^{<n}[x]$ that are not all zero polynomials. Let I denote the set of degree-bounded linear combinations of $\{z_i\}$, that is,*

$$I = \left\{ \sum_{i=1}^{t} a_i z_i \mid a_i \in \mathbb{Z}_q{}^{<n}[x] \right\}.$$

Moreover, let $d = \max_i \deg(z_i)$, and let $g = \gcd(z_1,\ldots,z_t)$. Then,

$$I = (g \cdot \mathbb{Z}_q[x]) \cap \mathbb{Z}_q{}^{<n+d}[x]$$

Moreover, for polynomials $(a_1,\ldots,a_t)$ sampled i.i.d. from $U(\mathbb{Z}_q{}^{<n}[x])$, the distribution $\left\{\sum_{i=1}^{t} a_i \cdot z_i\right\}$ is uniform on the set I.

Proof. Recall that $g = \gcd(z_1,\cdots,z_t)$ is some monic polynomial in $\mathbb{Z}_q[x]$ dividing each z_i. Therefore, the inclusion

$$I \subset (g \cdot \mathbb{Z}_q[x]) \cap \mathbb{Z}_q{}^{<n+d}[x]$$

is immediate. For the rest of the proof, we aim to show the opposite inclusion. We assume without loss of generality that all z_i are nonzero and prove the claim by induction on t.

We begin with a base case of $t = 2$ and further assume (at first) that $g = 1$. Fix polynomials (z_1, z_2) with $\gcd(z_1, z_2) = 1$, and assume WLOG that $d = \deg(z_1)$. We want to show that for every $\alpha \in \mathbb{Z}_q{}^{<n+d}[x]$, there exists a key (a_1, a_2) such that $h_{a_1,a_2}(z_1, z_2) = \alpha$. To do this, we write

$$\alpha = z_1 z_2 Q + R$$

for polynomials (Q, R) satisfying $\deg(Q) < n + d - \deg(z_1 z_2) = n - \deg(z_2)$, $\deg(R) < \deg(z_1 z_2)$. By the Chinese remainder theorem, we know that there exist polynomials s_1, s_2 satisfying $\deg(s_1) < \deg(z_2)$, $\deg(s_2) < \deg(z_1)$, and

$$s_1 z_1 + s_2 z_2 = R.$$

Then, choosing $a_1 = Qz_2 + s_1$ and $a_2 = s_2$, we see that $\deg(a_1) < n$, $\deg(a_2) < n$, and

$$a_1 z_1 + a_2 z_2 = z_1 z_2 Q + (s_1 z_1 + s_2 z_2) = \alpha,$$

as desired.

In the case that $\gcd(z_1, z_2) = g \neq 1$, for any $\alpha \in g\mathbb{Z}_q[x] \cap {\mathbb{Z}_q}^{<n+d}[x]$ write $\alpha = g\alpha'$ with $\deg(\alpha') < n + d - \deg(g)$. Since $\gcd(\frac{z_1}{g}, \frac{z_2}{g}) = 1$, we just showed that there exist a_1, a_2 with $\deg(a_i) < n$ and $a_1 \frac{z_1}{g} + a_2 \frac{z_2}{g} = \alpha'$, which implies that $a_1 z_1 + a_2 z_2 = g\alpha' = \alpha$. This completes the base case.

For the inductive step, consider any $t \geq 3$ and any polynomials $(z_1, \ldots z_t)$ with $d = \max_i \deg(z_i)$ and $g = \gcd(z_1, \ldots, z_t)$. We want to show that for any $\alpha \in g\mathbb{Z}_q[x] \cap {\mathbb{Z}_q}^{<n+d}[x]$, there exist polynomials $(a_1, \ldots, a_t)$ with $\deg(a_i) < n$ and $\sum_i a_i z_i = \alpha$.

We suppose without loss of generality that $\deg(z_1) = d$. Then, let $g' = \gcd(z_2, \ldots, z_t)$, and note that by the base case, there exist polynomials (a_1, a^*) such that $\deg(a_1) < n$, $\deg(a^*) < n$, and

$$a_1 z_1 + a^* g' = \alpha.$$

The base case applies because $\max(\deg(z_1), \deg(g')) = d$ and $\gcd(z_1, g') = g$. Now, further note that $\deg(a^* g') < n + \deg(g') \leq n + \max_{2 \leq i \leq t} \deg(z_i)$. Therefore, by the inductive hypothesis (applied to $(z_2, \ldots, z_t)$), there exist polynomials $(a_2, \ldots, a_t)$ such that $\deg(a_i) < n$ for all i, and

$$\sum_{i=2}^{n} a_i z_i = a^* g'.$$

This completes the inductive step.

Finally, we prove the distributional claim. Our reasoning follows the proof of ([14], Lemma 4.4). For every $\alpha \in I$, define the set

$$S_\alpha = \left\{ (a_1, \cdots, a_t) \in ({\mathbb{Z}_q}^{<n}[x])^t : \sum_{i=1}^{t} a_i z_i = \alpha \right\}.$$

By construction, the sets S_α for $\alpha \in I$ partition $({\mathbb{Z}_q}^{<n}[x])^t$. In order to prove the distributional claim, we only need to show that $|S_0| = |S_\alpha|$ for all $\alpha \in I$. To see this, note that for a given $\alpha \in I$, we have already shown that $S_\alpha \neq \emptyset$, so there exist $a_1', \cdots, a_t'$ such that $\deg(a_i') < n$ and $\sum_{i=1}^{t} a_i' z_i = b$. Then, the function $(a_i)_{i \leq t} \mapsto (a_i - a_i')_{i \leq t}$ is a bijection from S_α to S_0, proving that $|S_\alpha| = |S_0|$, as desired.

Having proved Lemma 10, we are ready to state and prove our variant of the leftover hash lemma.

Theorem 3. *Let χ be a distribution over $\mathbb{Z}_q$ and $\delta \in (0, 1)$ be such that $H_\infty(\chi) \geq \log(\frac{1}{\delta})$. Define the distribution $V := (\vec{a}, h_{\vec{a}}(\vec{r}))$ over $S = ({\mathbb{Z}_q}^{<n}[x])^t \times {\mathbb{Z}_q}^{<n+n'-1}[x]$, where $\vec{a} = (a_1, \ldots a_t)$ consists of i.i.d. samples from $U({\mathbb{Z}_q}^{<n}[x])$, and $\vec{r} = (r_1, \ldots, r_t)$ consists of i.i.d. samples from $\chi^{n'}[x]$.*

Then, for $n' \leq n$, *if* $\delta^t q = o(1)$,

$$\Delta(V, U(S)) = O\left(\delta^{\frac{t}{2}} q + \delta^{\frac{n't}{2}} q^{\frac{n+n'+1}{2}}\right).$$

In particular, for any $q = \mathsf{poly}(n)$, *if* $\delta^{-1} = \omega(1)$ *and* $n't/n = \Omega(\log n)$, *we have* $V \approx_s U(S)$.

Proof. By ([10], Claim 2) (i.e., by applying a generalized mean inequality), in order to prove that $\Delta(V, U(S)) \leq \epsilon$, it suffices to show that $\mathrm{CP}(V) \leq \frac{1+4\epsilon^2}{|S|}$; note that in our case, $|S| = q^{nt} \times q^{n+n'-1}$.

More precisely, let $\vec{a} = (a_i)_{i\in[t]}, \vec{a}' = (a'_i)_{i\in[t]}, \vec{r} = (r_i)_{i\in[t]}, \vec{r}' = (r'_i)_{i\in[t]}$ consist of i.i.d. samples from $U({\mathbb{Z}_q}^{<n}[x])$ and $\chi^{n'}[x]$ respectively. We want to show that

$$\mathrm{CP}(V) = \Pr\left[\vec{a} = \vec{a}' \wedge \sum_{i=1}^{t} a_i r_i = \sum_{i=1}^{t} a'_i r'_i\right] \leq \frac{1+4\epsilon^2}{q^{nt+n+n'-1}}.$$

We first partially evaluate the left-hand side of this inequality:

$$\begin{aligned} \Pr\left[\vec{a} = \vec{a}' \wedge \sum_{i=1}^{t} a_i r_i = \sum_{i=1}^{t} a'_i r'_i\right] &= \Pr\left[\vec{a} = \vec{a}' \wedge \sum_{i=1}^{t} a_i (r_i - r'_i) = 0\right] \\ &= q^{-nt} \Pr\left[\sum_{i=1}^{t} a_i (r_i - r'_i) = 0\right]. \qquad (1) \end{aligned}$$

Defining the random variable $\vec{v} = \vec{r} - \vec{r}'$, we then have

$$\begin{aligned} q^{-nt} \Pr\left[\sum_{i=1}^{t} a_i (r_i - r'_i) = 0\right] &= q^{-nt} \sum_{\vec{z}} \Pr\left[\vec{v} = \vec{z}\right] \Pr\left[\sum_{i=1}^{t} a_i z_i = 0\right] \\ &\leq q^{-nt} \left(\mathrm{CP}(\chi)^{n't} + \sum_{\vec{z}\neq 0} \frac{\Pr[\vec{v} = \vec{z}]}{|I(\vec{z})|}\right), \qquad (2) \end{aligned}$$

where $I(z) := \left\{\sum_{i=1}^{t} a_i z_i \mid a_i \in {\mathbb{Z}_q}^{<n}[x]\right\} = \gcd(z_1, \ldots, z_t)\mathbb{Z}_q[x] \cap {\mathbb{Z}_q}^{<n+\max_i \deg(z_i)}[x]$ as in Lemma 10. The last inequality follows from the distributional claim in Lemma 10.

To further simplify, we know by assumption that $\mathrm{CP}(\chi) \leq 2^{-H_\infty(\chi)} \leq \delta$. In addition, we group terms of the summation by the associated sets $I(z)$. That is, for every monic polynomial $g \in {\mathbb{Z}_q}^{<n'}[x]$ and degree $d < n'$, we define $I_{g,d} = g\mathbb{Z}_q[x] \cap {\mathbb{Z}_q}^{<n+d}[x]$ and obtain

$$\begin{aligned} q^{-nt} \left(\mathrm{CP}(\chi)^{n't} + \sum_{\vec{z}\neq 0} \frac{\Pr[\vec{v} = \vec{z}]}{|I(z)|}\right) &\leq q^{-nt} \left(\delta^{n't} + \sum_{\substack{g \text{ monic } \in {\mathbb{Z}_q}^{<n'}[x] \\ d<n'}} \Pr\left[I(\vec{v}) = I_{g,d}\right] \frac{1}{|I_{g,d}|}\right) \\ &\leq q^{-nt} \left(\delta^{n't} + \sum_{g,d} \Pr\left[I(\vec{v}) \subset I_{g,d}\right] \frac{1}{|I_{g,d}|}\right). \end{aligned} \qquad (3)$$

We next bound the probability that $I(\vec{v}) \subset I_{g,d}$ for any fixed g, d. To do this, we note by inspection that $I(\vec{v}) \subset I_{g,d}$ if and only if $v_i \in g\mathbb{Z}_q[x] \cap \mathbb{Z}_q{}^{<d+1}[x]$ for all i. For a fixed i, this occurs with probability

$$
\begin{aligned}
\Pr\left[v_i \in g\mathbb{Z}_q[x] \cap \mathbb{Z}_q{}^{<d+1}[x]\right] &= \Pr\left[v_i \in \mathbb{Z}_q{}^{<d+1}[x]\right] \Pr\left[v_i \in g\mathbb{Z}_q[x] \mid v_i \in \mathbb{Z}_q{}^{<d+1}[x]\right] \\
&= \mathrm{CP}(\chi)^{n'-d-1} \Pr\left[v_i \in g\mathbb{Z}_q[x] \mid v_i \in \mathbb{Z}_q{}^{<d+1}[x]\right] \\
&\leq \delta^{n'-d-1} \Pr\left[v_i \in g\mathbb{Z}_q[x] \mid v_i \in \mathbb{Z}_q{}^{<d+1}[x]\right] \qquad (4)
\end{aligned}
$$

In order to bound this probability, we define random variables w_i, w_i' to be drawn i.i.d. from $\chi^{d+1}[x]$ and compute

$$
\begin{aligned}
\Pr\left[v_i \in g\mathbb{Z}_q[x] \mid v_i \in \mathbb{Z}_q{}^{<d+1}[x]\right] &= \Pr\left[w_i - w_i' \in g\mathbb{Z}_q[x]\right] \\
&\leq \max_{\overline{w} \in \mathbb{Z}_q{}^{<\deg(g)}[x]} \Pr\left[w_i - \overline{w} \in g\mathbb{Z}_q[x]\right] \qquad (5)
\end{aligned}
$$

Fix an arbitrary $\bar{w}$. For a vector $v \in \mathbb{Z}_q^{d+1-\deg(g)}$ let T_v be set of polynomials $w_i \in \mathbb{Z}_q{}^{<d+1}[x]$ whose $(d+1-\deg(g))$ highest order coefficients are fixed to match v. Then, the "reduction mod g" map is a bijection from T_v to $\mathbb{Z}_q{}^{<\deg(g)}[x]$. Letting $\bar{w}^v$ denote the unique inverse of $\bar{w}$ in T_v and making use of the fact that $H_\infty(\chi) \geq \log(\frac{1}{\delta})$, we compute

$$
\begin{aligned}
\Pr\left[w_i - \overline{w} \in g\mathbb{Z}_q[x]\right] &= \sum_{v \in \mathbb{Z}_q^{d+1-\deg(g)}} \Pr[w_i \in T_v] \Pr[w_i = \overline{w}^v \mid w_i \in T_v] \\
&\leq \sum_{v \in \mathbb{Z}_q^{d+1-\deg(g)}} \Pr[w_i \in T_v] \delta^{\deg(g)} \\
&= \delta^{\deg(g)}. \qquad (6)
\end{aligned}
$$

Combining our calculations (equations (1)-(6)), we conclude that

$$\begin{aligned}
\mathrm{CP}(V) &\leq q^{-nt}\delta^{n't} + q^{-nt}\delta^{(n'-1)t} \sum_{\substack{g \text{ monic } \in \mathbb{Z}_q{}^{<n'}[x] \\ d<n'}} \delta^{\deg(g)-d} \frac{1}{|I_{g,d}|} \\
&= q^{-nt}\delta^{n't} + q^{-nt}\delta^{(n'-1)t} \sum_{\substack{g \text{ monic } \in \mathbb{Z}_q{}^{<n'}[x] \\ d<n'}} \delta^{(\deg(g)-d)t} \frac{1}{q^{n+d-\deg(g)}} \\
&= q^{-nt}\delta^{n't} + q^{-nt-n}\delta^{(n'-1)t} \sum_{\substack{d'=\deg(g)<n' \\ d<n'}} q^{d'} \left(\delta^{t} q\right)^{d'-d} \\
&= q^{-nt}\delta^{n't} + q^{-nt-n}\delta^{(n'-1)t} \sum_{\substack{d'<n' \\ d<n'}} \delta^{(d'-d)t} q^{2d'-d} \\
&\leq q^{-nt}\delta^{n't} + q^{-nt-n-n'+1}(1 + O(\delta^t q^2)) \\
&= q^{-nt-n-n'+1}\left(1 + O(\delta^t q^2 + \delta^{n't} q^{n+n'-1})\right),
\end{aligned}$$

where the final inequality follows from the assumption that $\delta^t q^2 = o(1)$. This completes the proof of Theorem 3.

For our application to IBE, we are interested in applying Theorem 3 in the case of a discrete Gaussian input distribution $D_{\mathbb{Z},\sigma}$. We now show that the hypothesis of Theorem 3 holds for sufficiently large σ.

Lemma 11. *Let* $\chi := D_{\mathbb{Z},\sigma}$ *and* $\chi_q := \chi \mod q$. *For* $\sigma = \mathsf{poly}(n), q = \omega(\sigma \log^{1/2} n), \sigma = \omega(1)$, *we have* $H_\infty(\chi_q) \geq \log(\frac{\sigma}{c})$*for some constant c.*

Proof. Since $q = \omega(\sigma \log^{1/2} n)$, only a negligible fraction of χ's probability mass "wraps around," i.e., is not contained in the interval $[-\frac{q}{2}, \frac{q}{2})$, so the min-entropy bound we proved about χ directly gives a min-entropy bound on χ_q.

In more detail, fix $\epsilon \in (0, 1/2)$ to be a small constant. By Lemma 2, $n_\epsilon(\mathbb{Z}) \leq c' \log(1 + \epsilon^{-1})$ for some constant c'. By Lemma 3 and our hypothesis, we see that

$$\Pr_{x \sim \chi}[|x| \geq q/2] = \mathsf{negl}(n).$$

Given this, we can compute

$$\begin{aligned}
2^{-H_\infty(\chi_q)} &= \max_{z \in \mathbb{Z}_q} \Pr_{x \sim \chi}[x \equiv z \pmod q] \\
&\leq \Pr_{x \sim \chi}[|x| \geq q/2] + \max_{z \in \mathbb{Z} \cap [-q/2, q/2]} Pr[x \equiv z \pmod q] \\
&\leq 2^{-H_\infty(\chi)} + \mathsf{negl}(n).
\end{aligned}$$

The bound $2^{-H_\infty}(\chi_q) \leq \frac{c}{\sigma}$ then follows from Lemma 5 applied to parameter $\delta = \sigma^{-1} c' \log(1 + \epsilon^{-1}) < 1/\sqrt{2}$; this parameter setting is possible since $\sigma = \omega(1)$.

5 Lattice Trapdoors for MP-LWE

In this section, we implement the "lattice trapdoors" paradigm of [8] for middle-product LWE. In particular, we show that the Micciancio-Peikert variant of lattice trapdoors [15] can be instantiated for MP-LWE.

In our setting, we want an algorithm TrapGen for generating random polynomials $(a_1, \ldots, a_{t'})$ along with a trapdoor td that allows for sampling polynomials (r_i) satisfying

$$\sum_{i=1}^{t'} a_i r_i = u$$

given any polynomial u (of the correct degree).

We briefly describe the method for generating (a_i). Let $t \leq t', d, n$ and distribution χ over $\mathbb{Z}_q$ be parameters to be defined later. For $(i,j) \in [t] \times [t'-t]$, we sample $a_i \xleftarrow{\$} {\mathbb{Z}_q}^{<n}[x]$ and $w_{i,j} \leftarrow \chi^d[x]$, and construct $a_{t+j} = c_j - \sum_{i \leq t} a_i \cdot w_{i,j}$, where $(c_j)_{j \in [t'-t]}$ is an analogue of matrix G in Definition 11. Note that $(a_i)_{i \leq t'} \in ({\mathbb{Z}_q}^{<n}[x])^t \times ({\mathbb{Z}_q}^{<n+d-1}[x])^{t'-t}$. We will choose d according to Theorem 3 to ensure that the distribution of each a_{t+j} is close to random. Finally, we will show that the trapdoor $\{w_{ij}\}$ can be used to implement the preimage sampling algorithm of [15] by considering the polynomials (a_i) as structured matrices, similarly to the Ring-LWE setting.

For the rest of this section, let $\tau := \lceil \log_2 q \rceil$. We first recall the notion of a "G-trapdoor" from [15].

Definition 11 ([15], Definition 5.2). *Let $G := I_k \otimes \begin{bmatrix} 1 & 2 & \cdots & 2^{\tau-1} \end{bmatrix} \in \mathbb{Z}_q^{k \times k\tau}$. Then, given a matrix $A \in \mathbb{Z}^{k \times (m+k\tau)}$, we say that a matrix $R \in \mathbb{Z}^{m \times k\tau}$ is a* ***G-trapdoor*** *for A if*

$$A \begin{bmatrix} R \\ I_{k\tau} \end{bmatrix} = G.$$

We make use of the following result in [15], Section 5.4, which states that G-trapdoors allow for efficient Gaussian preimage sampling in the style of [8].

Theorem 4 ([15], Theorem 5.5). *Let $G := I_k \otimes \begin{bmatrix} 1 & 2 & \cdots & 2^{\tau-1} \end{bmatrix} \in \mathbb{Z}_q^{k \times k\tau}$ and matrices $A \in \mathbb{Z}^{k \times (m+k\tau)}, R \in \mathbb{Z}^{m \times k\tau}$ be such that*

$$A \begin{bmatrix} R \\ I_{k\tau} \end{bmatrix} = G.$$

There exists an efficient algorithm $\mathcal{C} = (\mathcal{C}_1, \mathcal{C}_2)$ that operates as follows:

- *In the* ***offline phase****, $\mathcal{C}_1(A, R, \sigma)$ does some polynomial-time preprocessing on input (A, R, σ) and outputs a state* st.
- *In the* ***online phase****, $\mathcal{C}_2(\mathsf{st}, u)$, additionally given a vector u, samples from $D_{\Lambda_u^\perp(A),\sigma}$, as long as $\sigma \geq \omega(\sqrt{\log k})\sqrt{7(\sigma_1(R)^2+1)}$.*

Moreover, the runtime of $\mathcal{C}_2$ is the time to compute Rz for $z \in \mathbb{Z}^{k\tau}$ plus $\tilde{O}(m + k\tau)$.

We note that the proof of Theorem 4 given in [15] has a minor error that we correct in the Appendix. We now use Theorem 4 to instantiate lattice trapdoors for MP-LWE.

Theorem 5. *Suppose that* $q = \mathsf{poly}(n), d \leq n, dt/n = \Omega(\log n), \sigma = \omega(\log^2 n)\sqrt{ndt}$ *and* $\gamma = \frac{n+2d-2}{d}$ *is an integer. Then, there exist ppt algorithms* $(\mathsf{TrapGen}, \mathsf{SamplePre})$ *with the following properties.*

- $\mathsf{TrapGen}(1^n)$ *generates polynomials*

$$(a_1, \cdots, a_t, a_{t+1}, \cdots, a_{t+\gamma\tau}) \approx_s U((\mathbb{Z}_q{}^{<n}[x])^t \times (\mathbb{Z}_q{}^{<n+d-1}[x])^{\gamma\tau})$$

  together with a trapdoor td *that can be stored in* $O(n\tau t)$ *space.*
- $\mathsf{SamplePre}(\mathsf{td}, u)$ *that operates as follow:*
  - *In the* ***offline*** *phase, does some polynomial-time preprocessing with trapdoor* td *and parameter* σ, *and output a state* st.
  - *In the* ***online*** *phase, given state* st *and a syndrome* $u \in \mathbb{Z}_q{}^{<n+2d-2}[x]$, *outputs* $(r_i)_{i=1}^{t+\gamma\tau}$ *satisfying*

$$\sum_{i=1}^{t+\gamma\tau} a_i \cdot r_i = u$$

  in $\tilde{O}(nt)$ *time. Moreover, the output distribution of* (r_i) *is exactly the conditional distribution*

$$(D_{\mathbb{Z}^{2d-1},\sigma}[x])^t \times (D_{\mathbb{Z}^d,\sigma}[x])^{\gamma\tau} \mid \sum_{i=1}^{t+\gamma\tau} a_i \cdot r_i = u,$$

Proof. Let $\beta := \left\lceil \frac{\log_2 n}{2} \right\rceil$.

$\mathsf{TrapGen}$ *Algorithm:* We first describe $\mathsf{TrapGen}$ and prove that it outputs the right distribution of polynomials (a_i).

- For $(i, j) \in [t] \times [\gamma\tau]$, sample $a_i \xleftarrow{\$} \mathbb{Z}_q{}^{<n}[x]$ and $w_{i,j} \leftarrow \chi^d[x]$ where $\chi = U(\{-\beta, \cdots, \beta\})$. Since $\beta \ll q/2$, we can interpret samples from χ as elements of $\mathbb{Z}_q$.
- For all $j \in [\gamma\tau]$, define polynomials

$$u_j = \sum_{i=1}^{t} a_i \cdot w_{i,j}$$

$$a_{t+j} = c_j - u_j$$

  for $c_j \in \mathbb{Z}_q{}^{<n+d-2}[x]$ dependent only on j. Specifically, $c_j = 2^u x^{dv}$ for $j = v\tau + u + 1$ where $u \in \{0, \cdots, \tau - 1\}, v \in \{0, \cdots, \gamma - 1\}$.
- Output $(a_1, \ldots, a_{t+\gamma\tau})$ with associated trapdoor $\mathsf{td} = (w_{i,j})$.

We first note that the amount of space required to store $\mathsf{td} = (w_{i,j})$ is $O(d(\gamma\tau)t) = O(n\tau t)$, since $\gamma d = n + 2d - 2 \leq 3n$.

To see that the sampled polynomials $(a_1, \dots, a_{t+\gamma\tau})$ are statistically close to uniform, we apply our Leftover Hash Lemma (Theorem 3). In particular, $H_\infty(\chi) = \log(\frac{1}{\beta}) \geq \log\log n - 1$. Therefore, by Theorem 3,

$$(a_1, \cdots, a_t, u_1, \cdots, u_{\gamma\tau}) \approx_s U(\mathbb{Z}_q{}^{<n}[x]^t \times \mathbb{Z}_q{}^{<n+d-1}[x]^{\gamma\tau}),$$

and so

$$(a_i)_{i=1}^{t+\gamma\tau} = (a_1, \cdots, a_t, c_1 - u_1, \cdots, c_{\gamma\tau} - u_{\gamma\tau}) \approx_s U(\mathbb{Z}_q{}^{<n}[x]^t \times \mathbb{Z}_q{}^{<n+d-1}[x]^{\gamma\tau}).$$

$\mathsf{SamplePre}$ *Algorithm:* We next describe $\mathsf{SamplePre}$ using the algorithm from Theorem 4.

- Implicitly define matrices A, L by the following equations.

$$\begin{aligned}
\tilde{A} &= \left[T^{n,2d-1}(a_1)|\cdots|T^{n,2d-1}(a_t)\right] \\
\tilde{L} &= \begin{bmatrix} T^{d,d}(w_{1,1}) & \cdots & T^{d,d}(w_{1,\gamma\tau}) \\ \vdots & & \vdots \\ T^{d,d}(w_{t,1}) & \cdots & T^{d,d}(w_{t,\gamma\tau}) \end{bmatrix} \\
\Gamma(h) &= \left[T^{n+d-1,d}(h)|\cdots|T^{n+d-1,d}(h2^{\tau-1})\right] \\
G &= \left[\Gamma(1)|\Gamma(x^d)|\cdots|\Gamma(x^{(\gamma-1)d})\right] \\
I = I_{\gamma d\tau} &= \begin{bmatrix} T^{1,d}(1) & \cdots & \\ \cdots & & \cdots \\ & & T^{1,d}(1) \end{bmatrix} \\
A &= [\tilde{A}|G - \tilde{A}\tilde{L}] \\
L &= \begin{bmatrix} \tilde{L} \\ I \end{bmatrix}
\end{aligned} \tag{7}$$

so that $AL = G = I_{\gamma d} \otimes [1 \cdots 2^{\tau-1}]$, i.e., L is a G-trapdoor for A.
- Let $\mathbf{u} = T^{n+2d-2}(u) \in \mathbb{Z}_q^{n+2d-2}$ be the coefficient vector of u.
- Apply the algorithm from Theorem 4 (for $k = \gamma d = n + 2d - 2$) to sample $\mathbf{y}$ from $D_{\Lambda_{\mathbf{u}}^{\perp}(A),\sigma}$ where

$$\sigma = \omega(\sqrt{\log(\gamma d)})\beta\sqrt{7((\gamma d\tau)\cdot d \cdot t + 1)} = c\,\omega(\log^2 n)\sqrt{n \cdot (dt)},$$

for some constant c.
- Write $\mathbf{y}$ as $\begin{bmatrix} T^{2d-1,1}(r_1) \\ \vdots \\ T^{2d-1,1}(r_t) \\ T^{d,1}(r_{t+1}) \\ \vdots \\ T^{d,1}(r_{t+\gamma\tau}) \end{bmatrix}$, where $\deg(r_i) \begin{cases} < 2d - 1 \text{ for } i \in [t] \\ < d \text{ for } i \in \{t+1, \cdots, t+\gamma\tau\} \end{cases}$

– Output $(r_1, \dots, r_{t+\gamma\tau})$.

In order to analyze the correctness of $\mathsf{SamplePre}$, we first note that by construction, $\max_{i,j} |\tilde{L}_{ij}| \le \beta$, and so $\sigma_1(\tilde{L}) \le \beta\sqrt{(\gamma d\tau) \cdot (2d-1)t}$ by Lemma 6. Combined with Theorem 4 (for $k = \gamma d = n + 2d - 2$), this tells us that $\mathbf{y}$ is sampled from from $D_{\Lambda^{\perp}_{\mathbf{u}}(A),\sigma}$ where

$$\sigma = \omega(\sqrt{\log(\gamma d)})\beta\sqrt{7((\gamma d\tau) \cdot d \cdot t + 1)} = c\,\omega(\log^2 n)\sqrt{n \cdot (dt)}$$

for some constant c. Theorem 4 applies because of our parameter settings of $\gamma d = n + 2d - 2 \le 3n$ and $\tau = \theta(\log q) = \theta(\log n)$ for $q = \mathsf{poly}(n)$.

Moreover, by Lemma 7 and Equation (7),

$$\begin{aligned} \tilde{A}\tilde{L} &= \left[T^{n+d-1,d}(u_1)| \cdots |T^{n+d-1,d}(u_{\gamma\tau})\right], \text{ and} \\ A &= \left[T^{n,2d-1}(a_1)| \cdots |T^{n,2d-1}(a_t)|T^{n+d-1,d}(a_{t+1})| \cdots |T^{n+d-1,d}(a_{t+\gamma\tau})\right], \end{aligned} \tag{8}$$

and so by Equation (8) and Lemma 7

$$A\mathbf{y} = T^{n+2d-2,1}(\sum_{i=1}^{t+\gamma\tau} a_i \cdot r_i). \tag{9}$$

Thus, $\mathbf{y} \in \Lambda^{\perp}_{\mathbf{u}}(A)$ if and only if $\sum_{i=1}^{t+\gamma\tau} a_i \cdot r_i = u$.

To prove the claim about distribution of $\mathbf{r} = (r_i)$, we note that the columns of A generate $\mathbb{Z}^{n+2d-2}$ since (1) the columns of G generate $\mathbb{Z}^{n+2d-2}$ and (2) $AL = G$. Hence, there exists $\mathbf{y}^*$ such that $A\mathbf{y}^* = \mathbf{u}$. Then, Lemma 5.2 in [8] applies, allowing us to conclude that the distribution of $\mathbf{y}$ sampled by our algorithm is exactly

$$D_{\Lambda^{\perp}_{\mathbf{u}}(A),\sigma} \equiv \mathbf{y}^* + D_{\Lambda^{\perp}(A),\sigma,-\mathbf{y}^*} \equiv D^{(2d-1)t+d\gamma\tau}_{\mathbb{Z},\sigma} \mid A\mathbf{y} = \mathbf{u}.$$

To see that the first equality of distributions holds, note that the two distributions have the same support (i.e., $t + \Lambda^{\perp}(A) = \Lambda^{\perp}_{\mathbf{u}}(A)$), and for all $x \in \Lambda^{\perp}_{\mathbf{u}}(A)$,

$$D_{\Lambda^{\perp}_{\mathbf{u}}(A),\sigma}(x) = \frac{\rho_\sigma(x)}{\rho_\sigma(\Lambda^{\perp}_{\mathbf{u}}(A))} = \frac{\rho_\sigma(x - t + t)}{\rho_\sigma(\Lambda^{\perp}(A)) + t} = D_{\Lambda^{\perp}(A),\sigma,-t}(x - t).$$

Thus, the conditional distribution of $\mathbf{r}$ is as claimed. Finally, we analyze the runtime of $\mathsf{SamplePre}$'s online phase, which is precisely the runtime of $\mathcal{C}_2$. Computing $\tilde{L}z$ for $z \in \mathbb{Z}_q^{\gamma d\tau}$ can be performed, using polynomial multiplication, in $O((d\log d)t\gamma\tau) = \tilde{O}(nt)$ time; this bound uses the fact that $\gamma d \le 3n$, $\log d \le \log n$ and $\tau = \Theta(\log n)$.

6 New Encryption Schemes from Middle-Product LWE

In this section, we describe how to build a "Dual Regev"-style public-key encryption scheme, as well as an identity-based encryption scheme, whose security is based on the hardness of MP-LWE. As in [8], our IBE scheme is constructed by combining the Dual Regev scheme with lattice trapdoors as constructed in Section 5.

6.1 Middle Product Dual Regev Encryption

Unless otherwise stated, the following parameters are positive integers.

Let $q = q(n)$ be a prime, $\tau := \lceil \log_2 q \rceil, n, d, k$ be such that $\gamma = \frac{n+2d-2}{d} \in \mathbb{N}$ and $2d + k \leq n$. Let $t > 0, t' = t + \gamma\tau$. Let $\chi := \lfloor D_{\alpha \cdot q} \rceil$ be the distribution over $\mathbb{Z}$ in which $\epsilon \leftarrow D_{\alpha \cdot q}$ is sampled and then rounded to the nearest integer.

Finally, let $\sigma \in \mathbb{R}_{>0}$ be a parameter to be specified later. We then define a public-key encryption scheme with message space $\mathcal{M} = \{0,1\}^{<k+1}[x]$.

- **Key Generation**: $\mathsf{KeyGen}(1^n)$ operates as follows.
  - For $1 \leq i \leq t$, sample $a_i \xleftarrow{\$} {\mathbb{Z}_q}^{<n}[x]$, $r_i \leftarrow D_{\mathbb{Z}^{2d-1},\sigma}[x]$;
  - For $t+1 \leq i \leq t'$, sample $a_i \xleftarrow{\$} {\mathbb{Z}_q}^{<n+d-1}[x]$, $r_i \leftarrow D_{\mathbb{Z}^d,\sigma}[x]$.
  - Compute $u = \sum_{i=1}^{t'} a_i r_i$ and output $\mathsf{pk} := (a_1, \ldots, a_{t'}, u)$; $\mathsf{sk} := (r_1, \ldots, r_{t'})$
- **Encryption:** $\mathsf{Enc}(\mathsf{pk} = ((a_i)_{i \leq t'}, u), \mu)$ operates as follows.
  - Sample $s \xleftarrow{\$} {\mathbb{Z}_q}^{<n+2d+k-1}[x]$
  - For $1 \leq i \leq t$, sample $e_i \leftarrow \chi^{2d+k}[x]$, and compute $b_i = a_i \odot_{2d+k} s + 2e_i$
  - For $t+1 \leq i \leq t'$, sample $e_i \leftarrow \chi^{d+k+1}[x]$, and compute $b_i = a_i \odot_{d+k+1} s + 2e_i$
  - Sample $e' \leftarrow \chi^{k+1}[x]$, and compute $c_1 = \mu + u \odot_{k+1} s + 2e'$
  - Output $c = (c_1, (b_i)_{i \leq t'})$.
- **Decryption:** $\mathsf{Dec}(\mathsf{sk} = (r_i)_{i \leq t}, c = (c_1, (b_i)_{i \leq t'})$ outputs $(c_1 - \sum_{i=1}^{t'} b_i \odot_{k+1} r_i \mod q) \mod 2$.

Lemma 12. *For $\alpha^{-1} > (4\omega(\log n)\sigma K + 1)$ where $K := t(2d-1) + \gamma\tau d$, the scheme satisfies $(1 - \mathsf{negl}(n))$-correctness.*

Proof. We want to show that $\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, \mu)) = 1$ with probability $1 - \mathsf{negl}(n)$ over the randomness of KeyGen and Enc. Consider a random key pair $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^n)$ and ciphertext $c = (c_1, (b_i)_{i \leq t'}) \leftarrow \mathsf{Enc}(\mathsf{pk}, \mu)$. By Lemma 1 (the quasi-associative law for middle products),

$$\begin{aligned} c_1 &= \mu + \sum_{i=1}^{t'} (r_i \cdot a_i) \odot_{k+1} s + 2e' \\ &= \mu + \sum_{i=1}^{t} r_i \odot_{k+1} (a_i \odot_{2d+k} s) + \sum_{i=t+1}^{t'} r_i \odot_{k+1} (a_i \odot_{d+k+1} s). \end{aligned}$$

Therefore, we see that

$$c_1 - \sum_{i=1}^{t'} b_i \odot_{k+1} r_i = \mu + 2(e' - \sum_{i=1}^{t'} r_i \odot_{k+1} e_i).$$

We conclude that if $\left|\left|\mu + 2(e' - \sum_{i=1}^{t'} r_i \odot_{k+1} e_i)\right|\right|_\infty < q/2$, then $\mathsf{Dec}(\mathsf{sk}, c)$ will indeed output the message μ.

To complete the proof of correctness, we want to bound the coefficients of $\sum_{i=1}^{t'} r_i \odot_{k+1} e_i$. The coefficient of x^ℓ in $r_i \odot_{k+1} e_i$ is

$$\sum_{w \in [0, \deg(r_i)] \cap [\ell+k-\deg(e_i), z+k]} (\mathbf{r}_i)_w (\mathbf{e}_i)_{\ell+k-w}.$$

Using our discrete Gaussian tail inequality (see Lemma 3) and a union bound, we obtain the following bounds on $||\mathbf{r}_i||_\infty$ and $||\mathbf{e}_i||_\infty$:

$$\Pr[||\mathbf{r}_i||_\infty > \omega(\sqrt{\log n})\sigma] = \mathsf{negl}(n).$$

$$\Pr[||\mathbf{e}_i||_\infty > \omega(\sqrt{\log n})\alpha q] = \mathsf{negl}(n).$$

Thus, again by union bound, except with $\mathsf{negl}(n)$ probability

$$\left|\left| e' - \sum_{i=1}^{t'} r_i \odot_{k+1} e_i \right|\right|_\infty < K(\omega(\sqrt{\log n})\sigma)(\omega(\sqrt{\log n})\alpha \cdot q) + (\omega(\sqrt{\log n})\alpha \cdot q).$$

for $K := t(2d-1) + \gamma\tau d \geq \sum_{i=1}^{t'}(\deg(r_i)+1)$. Picking $\alpha < (4\omega(\log n)\sigma K + 1)^{-1}$, the above is less than $q/4$ and so the scheme is $(1 - \mathsf{negl}(n))$-correct.

Theorem 6. *Assume that* $\sigma = \omega(1), dt/n = \Omega(\log n)$, q *is a prime polynomial in* n, $q = \Omega(\alpha^{-1} n^{1/2+1/2+c})$ *and* $q = \omega(\log^{1/2} n)\sigma$. *The scheme is semantically secure assuming* $\mathsf{PLWE}^{(f)}_{q, D_{\alpha' \cdot q}}$ *is hard for some polynomial* f *such that the constant coefficient of* f *is coprime with* $q, \deg(f) \in [2d+k, n], \mathrm{EF}(f) = O(n^c)$ *and error* $\alpha' = \Omega(\sqrt{\deg(f)}/q)$.

Proof. By Theorem 3, Lemma 11 and hypothesis on σ and dt, we have:

$$\left((a_i)_{i=1}^t, \sum_{i=1}^t a_i \cdot r_i\right)_{\substack{a_i \xleftarrow{\$} \mathbb{Z}_q{}^{<n}[x] \\ r_i \leftarrow D_{\mathbb{Z}^{2d-1},\sigma}[x]}} \approx_s \left((a_i)_{i=1}^t, u'\right)_{\substack{a_i \xleftarrow{\$} \mathbb{Z}_q{}^{<n}[x] \\ u' \xleftarrow{\$} \mathbb{Z}_q{}^{<n+2d-2}[x]}}$$

Since an honestly generated public key has the form $\mathsf{pk} = (a_1 \ldots, a_{t'}, u)$ for $u = \sum_{i=1}^t a_i \cdot r_i + \sum_{i=t+1}^{t'} a_i \cdot r_i$, we see that pk is computationally indistinguishable from a public key $\widetilde{\mathsf{pk}}$ of the form

$$\widetilde{\mathsf{pk}} = (a_1, \ldots, a_{t'}, u), u \xleftarrow{\$} \mathbb{Z}_q{}^{<n+2d-2}[x].$$

Thus, we see that for any message μ, we have

$$\left(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, \mu)\right) \approx_s \left(\widetilde{\mathsf{pk}}, \mathsf{Enc}(\widetilde{\mathsf{pk}}, \mu)\right).$$

Moreover, we have

$$\left(\widetilde{\mathsf{pk}}, \mathsf{Enc}(\widetilde{\mathsf{pk}}, \mu)\right) \approx_c \left(\widetilde{\mathsf{pk}}, \mathsf{Enc}(\widetilde{\mathsf{pk}}, 0)\right)$$

assuming the hardness of (degree-parametrized) $\mathsf{MPLWE}_{q,n+2d+k,\mathbf{d},\lfloor D_{\alpha\cdot q}\rceil}$ with degree vector

$$\mathbf{d}_i = \begin{cases} 2d+k, \text{ if } i \in [t] \\ d+k+1, \text{ if } t+1 \leq i \leq t' \\ k+2, \text{ if } i = t'+1. \end{cases}$$

The hardness of $\mathsf{MPLWE}_{q,n+2d+k,\mathbf{d},\lfloor D_{\alpha\cdot q}\rceil}$ follows from the hardness of $\mathsf{MPLWE}_{q,n+2d+k,\mathbf{d},D_{\alpha\cdot q}}$ via a standard reduction that maps $(a,b) \in \mathbb{Z}_q[x] \times \mathbb{R}_q[x]$ to $(a, \lceil b \rfloor)$, where $\lceil b \rfloor$ is the polynomial obtained by rounding every coefficient of b to the nearest integer. Finally, Theorem 2 tells us that $\mathsf{MPLWE}_{q,n+2d+k,\mathbf{d},D_{\alpha\cdot q}}$ is hard assuming the hardness of $\mathsf{PLWE}^{(f)}_{q,D_{\alpha'\cdot q}}$, for $\alpha \cdot q = \Omega(n^{1/2+1/2+c}) \geq \alpha' \cdot q\sqrt{\frac{n+2d+k}{2}}n^c$. This completes the proof of semantic security.

6.2 IBE in the Random Oracle model

We construct an IBE scheme in the random oracle model by combining our "Dual Regev" scheme (Section 6.1) with our MP-LWE lattice trapdoors (Section 5). The IBE construction is essentially identical to that of [8]; we give an explicit description for completeness. Let the set of identity be $\mathcal{I} = \mathbb{Z}_q^{n+2d-2}$. We assume the parameters are chosen such that Theorem 5 holds. Use algorithm $\mathsf{TrapGen}$ to generate mpk:=$(a_i)_{i=1}^{t'}$ and msk:=$\tilde{L}$. Given an identity id, interpret it as an element $u \in \mathbb{Z}_q{}^{<n+2d-2}[x]$ and use algorithm $\mathsf{SamplePre}$ to generate $\mathsf{sk}_{id} := (r_i)_{i=1}^{t'}$ such that $\sum_{i=1}^{t'} a_i \cdot r_i = u$. Then use the Dual Regev scheme with public key $\mathsf{pk} := ((a_i)_{i=1}^{t'}, u)$ and secret key $\mathsf{sk} := (r_i)_{i=1}^{t'}$ for encryption/decryption of message.

- **Setup:** The setup algorithm $\mathsf{IBESetup}$ (on input 1^n) calls $\mathsf{TrapGen}(1^n)$, obtaining polynomials $(a_1, \ldots, a_{t'})$ along with a trapdoor td. It outputs master public key $\mathsf{mpk} = (a_1, \ldots, a_{t'})$ and master secret key $\mathsf{msk} = \mathsf{td}$.
- **Key Extraction:** The secret key extraction algorithm $\mathsf{IBEExtract}$, given msk and an identity id, calls $\mathsf{SamplePre}(\mathsf{td}, H(\mathsf{id}))$, where $H(\cdot)$ is modelled as a random oracle. It outputs $\mathsf{sk}_{\mathsf{id}} = (r_1, \ldots, r_{t'})$, the output of $\mathsf{SamplePre}$.
- **Encryption:** The encryption algorithm Enc, given the master public key $\mathsf{mpk} = (a_1, \ldots, a_{t'})$, an identity id, and a message μ, computes $u = H(\mathsf{id})$ and outputs a ciphertext $c \leftarrow \mathsf{DualRegev.Enc}(\mathsf{pk}_{\mathsf{id}}, \mu)$ (using the Dual Regev encryption algorithm) for $\mathsf{pk}_{\mathsf{id}} = (a_1, \ldots, a_{t'}, u)$
- **Decryption:** The decryption algorithm Dec, given the secret key $\mathsf{sk}_{\mathsf{id}} = (r_1, \ldots, r_{t'})$ and a ciphertext c, outputs $\mathsf{DualRegev.Dec}(\mathsf{sk}_{\mathsf{id}}, c)$.

Theorem 7. *Assume the parameters are picked as in Theorem 5 and Theorem 6 (so that the Dual Regev Scheme is correct and semantically secure). Then the above IBE scheme is correct and CPA-secure in the random oracle model.*

Proof. See ([8], Theorem 7.2).

Remark 2 (Efficiency). Pick $d, k = \Theta(n), t = \log n$ and σ, α^{-1}, q satisfying bounds in Theorems 5 and 6. By construction, the schemes in subsections 6.1 and 6.2 have key size and ciphertext size $\tilde{O}(n)$. We show that encryption and decryption algorithms in these schemes take $\tilde{O}(n)$ time. As in [21], products and middle products of polynomials can be computed in $\tilde{O}(n)$ time using FFT-based techniques [22, 9]. By doing some preprocessing, sampling from $\chi = \lfloor D_{\alpha \cdot q} \rceil$ can be done in quasi-constant time via table look-up as in [15]. Thus, encryption and decryption in our Dual-Regev like public key scheme and IBE scheme take $\tilde{O}(n)$ time; since the message is of size $k = \Theta(n)$, runtime per encrypted bit is $\tilde{O}(1)$.

6.3 IBE in the Standard model

[2, 7] present IBE schemes secure in the standard model from the same framework of lattice trapdoors and dual-Regev encryption. A simplified version of one construction is presented in [3], Section 3. We give a brief summary of [3]'s IBE construction, and sketch how to adapt it to the MP-LWE setting.

Suppose we have an identity space $\{0,1\}^\ell$. Set $m = O(n \log n)$. In $\mathsf{IBESetup}$, the [3] scheme samples a random matrix $A \in \mathbb{Z}_q^{n \times m}$ together with trapdoor T_A, as well as random matrices $H_{i,b} \in \mathbb{Z}_q^{n \times m}$ for $i \in [\ell]$ and $b \in \{0,1\}$. The public key is $(A, (H_{i,b})_{(i,b) \in [\ell] \times \{0,1\}}, u_0)$ where u_0 is a random vector in $\mathbb{Z}_q^n$. The master secret key is T_A. The key extraction algorithm $\mathsf{IBEExtract}$, given an identity $\mathsf{id} = \mathsf{id}_1 \cdots \mathsf{id}_\ell$, assembles $H_{\mathsf{id}} = H_{1,\mathsf{id}_1} | \cdots | H_{\ell,\mathsf{id}_\ell} \in \mathbb{Z}_q^{n \times \ell m}$ as the concatenation of ℓ matrices. It then samples random vectors $r_i \in \mathbb{Z}_q^m$, and constructs a vector $r = (r_i)_{i \in \ell} \in \mathbb{Z}_q^{\ell m}$. Finally, the trapdoor T_A is used to sample preimages $e \in \mathbb{Z}_q^m$ satisfying $A\, e = u_0 + H_{\mathsf{id}}\, r$, i.e. $\begin{bmatrix} A \mid -H_{\mathsf{id}} \end{bmatrix} \begin{bmatrix} e \\ r \end{bmatrix} = u_0$, yielding a secret key $\mathsf{sk}_{\mathsf{id}} = (e, r)$. Encryption and Decryption then proceed as in Dual Regev encryption.

[3] proves that their scheme is selective-ID secure in the standard model. Selective-ID security is defined by a game similar to that in Section 2.1, except that the adversary generates the challenge identity id^* *before* seeing the public parameters of the scheme. [2]'s proof of selective-ID security relies on replacing each random matrix $H_{i,b}$ with an indistinguishable matrix $H'_{i,b}$ equipped with trapdoor $T_{i,b}$. Then, given an extraction query id that differs from the challenge id^*, letting i denote an index on which $\mathsf{id}_i \neq \mathsf{id}_i^*$, the trapdoor T_{i,id_i} can be used to sample r_i such that $H'_{i,\mathsf{id}_i}\, r_i = A\, e - \sum_{j \neq i} H'_{j,\mathsf{id}_j}\, r_j - u_0$, where e and r_j are sampled randomly from $\mathbb{Z}_q^m$, to produce a secret key $\mathsf{sk}_{\mathsf{id}} = (e, r)$. Crucially, sampling (e, r) using trapdoor T_{i,id_i} is (statistically) indistinguishable from the honest key extraction procedure (that uses T_A).

Our MP-LWE Dual Regev encryption scheme, combined with the lattice trapdoors of Theorem 5, can be used to create a standard model IBE scheme analogous to the one just described. Specifically, we replace the matrix A and its trapdoor T_A with tuple $(a_j)_{j \leq t}$ of $t = \tilde{O}(1)$ polynomials and its trapdoor as generated by Theorem 5. We replace each matrix $H_{i,b}$ with tuple $\bar{h}^{(i,b)} = (h_j)_{j \leq t}^{(i,b)}$ of random polynomials, and replace random vector u_0 with a random polynomial. Theorem 5 allows us to replace any particular $\bar{h}^{(i,b)}$ with $\tilde{h}^{(i,b)}$ that is equipped with a trapdoor $T_{i,b}$, and our $\mathsf{SamplePre}$

algorithm guarantees the same $(T_A, T_{i,b})$ indistinguishability that was leveraged by [3].

To summarize, this allows for an IBE scheme in the standard model based on MP-LWE with efficiency gains of $\tilde{O}(n)$ over the [3] scheme.

References

[1] Abdalla, M., Bellare, M., Catalano, D., Kiltz, E., Kohno, T., Lange, T., Malone-Lee, J., Neven, G., Paillier, P., Shi, H.: Searchable encryption revisited: consistency properties, relation to anonymous ibe, and extensions. In: Shoup, V. (ed.) Advances in Cryptology – CRYPTO 2005. pp. 205–222. Springer Berlin Heidelberg, Berlin, Heidelberg (2005)

[2] Agrawal, S., Boneh, D., Boyen, X.: Efficient lattice (h)ibe in the standard model. In: Proceedings of the 29th Annual International Conference on Theory and Applications of Cryptographic Techniques. pp. 553–572. EUROCRYPT'10, Springer-Verlag, Berlin, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13190-5_28, http://dx.doi.org/10.1007/978-3-642-13190-5_28

[3] Agrawal, S., Boyen, X.: Identity-based encryption from lattices in the standard model (2009)

[4] Bai, S., Langlois, A., Lepoint, T., Stehlé, D., Steinfeld, R.: Improved security proofs in lattice-based cryptography: Using the rényi divergence rather than the statistical distance. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 3–24. Springer (2015)

[5] Bellare, M., Desai, A., Jokipii, E., Rogaway, P.: A concrete security treatment of symmetric encryption. In: Proceedings 38th Annual Symposium on Foundations of Computer Science. pp. 394–403. IEEE (1997)

[6] Boneh, D., Franklin, M.K.: Identity-based encryption from the weil pairing. In: Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology. pp. 213–229. CRYPTO '01, Springer-Verlag, London, UK, UK (2001), http://dl.acm.org/citation.cfm?id=646766.704155

[7] Cash, D., Hofheinz, D., Kiltz, E., Peikert, C.: Bonsai trees, or how to delegate a lattice basis. In: Gilbert, H. (ed.) Advances in Cryptology – EUROCRYPT 2010. pp. 523–552. Springer Berlin Heidelberg, Berlin, Heidelberg (2010)

[8] Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: Proceedings of the fortieth annual ACM symposium on Theory of computing. pp. 197–206. ACM (2008)

[9] Hanrot, G., Quercia, M., Zimmermann, P.: The middle product algorithm i. Applicable Algebra in Engineering, Communication and Computing **14**(6), 415–438 (2004)

[10] Impagliazzo, R., Zuckerman, D.: How to recycle random bits. In: 30th Annual Symposium on Foundations of Computer Science. pp. 248–253. IEEE (1989)

[11] Lyubashevsky, V.: Digital signatures based on the hardness of ideal lattice problems in all rings. In: Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part II. pp. 196–214 (2016)

[12] Lyubashevsky, V., Micciancio, D.: Generalized compact knapsacks are collision resistant. In: Wegener, I., Sassone, V., Preneel, B. (eds.) Proceedings of the 33rd international colloquium on automata, languages and programming - ICALP 2006. Lecture Notes in Computer Science, vol. 4052, pp. 144–155. Springer-Verlag, Venice, Italy (Jul 2006)

[13] Lyubashevsky, V., Peikert, C., Regev, O.: On ideal lattices and learning with errors over rings. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 1–23. Springer (2010)

[14] Micciancio, D.: Generalized compact knapsacks, cyclic lattices, and efficient one-way functions from worst-case complexity assumptions. In: Proceedings of the 43rd Symposium on Foundations of Computer Science. pp. 356–365. FOCS '02, IEEE Computer Society, Washington, DC, USA (2002), http://dl.acm.org/citation.cfm?id=645413.652130

[15] Micciancio, D., Peikert, C.: Trapdoors for lattices: Simpler, tighter, faster, smaller. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 700–718. Springer (2012)

[16] Micciancio, D., Regev, O.: Worst-case to average-case reductions based on gaussian measures. SIAM Journal on Computing **37**(1), 267–302 (2007)

[17] Peikert, C., Regev, O., Stephens-Davidowitz, N.: Pseudorandomness of ring-lwe for any ring and modulus. In: Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing. pp. 461–473. ACM (2017)

[18] Peikert, C., Rosen, A.: Efficient collision-resistant hashing from worst-case assumptions on cyclic lattices. In: Proceedings of the Third Conference on Theory of Cryptography. pp. 145–166. TCC'06, Springer-Verlag, Berlin, Heidelberg (2006). https://doi.org/10.1007/11681878_8, http://dx.doi.org/10.1007/11681878_8

[19] Peikert, C., Vaikuntanathan, V., Waters, B.: A framework for efficient and composable oblivious transfer. In: Wagner, D. (ed.) Advances in Cryptology – CRYPTO 2008. pp. 554–571. Springer Berlin Heidelberg, Berlin, Heidelberg (2008)

[20] Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Proceedings of the thirty-seventh annual ACM symposium on Theory of computing. pp. 84–93. ACM (2005)

[21] Roşca, M., Sakzad, A., Stehlé, D., Steinfeld, R.: Middle-product learning with errors. In: Annual International Cryptology Conference. pp. 283–297. Springer (2017)

[22] Shoup, V.: Efficient computation of minimal polynomials in algebraic extensions of finite fields. In: In Proceedings of the 1999 International Symposium on Symbolic and Algebraic Computation (Vancouver, BC. Citeseer (1999)

[23] Stehlé, D., Steinfeld, R., Tanaka, K., Xagawa, K.: Efficient public key encryption based on ideal lattices. In: Matsui, M. (ed.) Advances in Cryptology – ASIACRYPT 2009. pp. 617–635. Springer Berlin Heidelberg, Berlin, Heidelberg (2009)

Appendix

We describe a minor correction to the proof of ([15], Theorem 5.5).

In [15], it is mistakenly claimed (see Section 2.1) that for positive semi-definite $B \geq A \geq 0$, the inequality $A^+ \geq B^+$ holds. This is not true in general. For example, when B is positive definite (that is, $B > 0$), we have $B^+ = B^{-1} > 0$; if $A^+ \geq B^+$ then $A^+ > 0$ so $A > 0$ (a contradiction if A is not invertible).

However, the proof of ([15], Theorem 5.5) can be modified to avoid using the mistaken claim. The relevant setting is as follows: consider a matrix of the form

$$\Sigma_3 = (\Sigma_y^+ + \Sigma_p^+)^+$$

such that $\Sigma_y = 2\begin{bmatrix}R\\I\end{bmatrix}\begin{bmatrix}R^T & I\end{bmatrix}$ and $\Sigma_p > 2\begin{bmatrix}R\\I\end{bmatrix}\begin{bmatrix}R^T & I\end{bmatrix}$.[6] We want to prove that $\Sigma_3 \geq \begin{bmatrix}R\\I\end{bmatrix}\begin{bmatrix}R^T & I\end{bmatrix}$ (see p.29)

To prove this, we write $2\begin{bmatrix}R\\I\end{bmatrix}\begin{bmatrix}R^T & I\end{bmatrix} = Q\begin{bmatrix}D & 0\\0 & 0\end{bmatrix}Q^T$ where Q is orthogonal and D is diagonal matrix of positive entries. Then, there exists some small $\epsilon > 0$ s.t. $\Sigma_p \geq Q\begin{bmatrix}D & 0\\0 & \epsilon I\end{bmatrix}Q^T > 0$. Thus,

$$0 < \Sigma_p^+ = \Sigma_p^{-1} \leq Q\begin{bmatrix}D^{-1} & 0\\0 & \epsilon^{-1}I\end{bmatrix}Q^T = \frac{1}{2}\left(\begin{bmatrix}R\\I\end{bmatrix}\begin{bmatrix}R^T & I\end{bmatrix}\right)^+ + \begin{bmatrix}0 & 0\\0 & \epsilon^{-1}I\end{bmatrix}$$

and so

$$0 < \Sigma_p^+ + \Sigma_y^+ \leq \left(\begin{bmatrix}R\\I\end{bmatrix}\begin{bmatrix}R^T & I\end{bmatrix}\right)^+ + \begin{bmatrix}0 & 0\\0 & \epsilon^{-1}I\end{bmatrix} = Q\begin{bmatrix}2D^{-1} & 0\\0 & \epsilon^{-1}I\end{bmatrix}Q^T.$$

We conclude that

$$(\Sigma_y^+ + \Sigma_p^+)^+ = (\Sigma_y^+ + \Sigma_p^+)^{-1} \geq Q\begin{bmatrix}\frac{1}{2}D & 0\\0 & \epsilon I\end{bmatrix}Q^T \geq \begin{bmatrix}R\\I\end{bmatrix}\begin{bmatrix}R^T & I\end{bmatrix}.$$

[6] Our assumptions on Σ_y and Σ_p are slightly different from those of [15]; these minor modifications are without loss of generality with respect to the application to Gaussian sampling but necessary for the proof to go through.