Exploring General Cyclotomic Rings in Torus-Based Fully Homomorphic Encryption: Part I - Prime Power Instances*

P. Chartier† M. Koskas‡ and M. Lemou§

May 28, 2025

Abstract

In this article, we delve into the domain of fully homomorphic encryption over the torus, focusing on the algebraic techniques required for managing polynomials within cyclotomic rings defined by prime power indices. Our study encompasses essential operations, such as modulo reduction, efficient homomorphic evaluation of trace operators, blind extraction, and the blind rotation pivotal to the bootstrapping process, all redefined within this mathematical context. Through the extensive application of duality theory and trace operators in general cyclotomic rings or fields, we systematize and enhance these operations, introducing a simplified formulation of bootstrapping alongside an effective packing strategy. This investigation serves as an initial step toward addressing the broader case of composite cyclotomic indices, which we expect will open up new avenues for cryptographic applications and functionalities.

Keywords: fully homomorphic encryption, residue number system, trace operator, extraction, bootstrapping.

Contents

*This work was partly conducted while the first and last authors were affiliated with Ravel Technologies and is protected by Ravel Technologies patents.

†INRIA-IRMAR-University of Rennes, Campus de Beaulieu, 35042 Rennes Cedex
‡Ravel Technologies, 75 rue de Richelieu, 75002 Paris
§CNRS-IRMAR-University of Rennes, Campus de Beaulieu, 35042 Rennes Cedex

1 Introduction

Non-power of two cyclotomic polynomials, which pertain to cyclotomic rings indexed by integers M that are not restricted to the form $M = 2^k$, carry significant implications for fully homomorphic encryption (FHE) systems. Non-power of two cyclotomic polynomials Φ_M are associated with a wider variety of primitive roots of unity. Generally speaking, this diversity allows for richer algebraic structures and facilitates computations on more complex data types, thereby enhancing the versatility of Fully Homomorphic Encryption (FHE) schemes. Specifically, we highlight five key advantages that we believe strongly justify the significance of our current work. Several of these points have already been emphasized by Vadim Lyubashevsky, Peikert, and Regev in [30], as well as by Joye and Walter in [25].

- **Increased Input Space:** Utilizing cyclotomic rings beyond powers of two can significantly expand the plaintext space available for bootstrapping operations. This

makes it feasible to encrypt and perform homomorphic computations on a greater range of data, improving the overall utility of the encryption scheme. In the power-of-two case, algebraic constraints effectively halves the size of messages that can be accurately bootstrapped. In general, the proportion of the usable plaintext space is limited by the ratio N/M, where N is the degree of the cyclotomic polynomial. For $M = t^\alpha$, we have $N = (t-1)t^{\alpha-1}$, which yields $N/M = 1 - \frac{1}{t}$. This ratio increases with larger values of t, indicating improved efficiency in the available plaintext space. Here, we will further demonstrate how to completely eliminate the so-called "nega-cyclicity" condition involved in bootstrapping by choosing parameter t adequately and without any additional cost.

- **Flexibility in Parameter Choices:** Non-power of two cyclotomic polynomials enable the customization of FHE parameters to better align with specific applications or security requirements. For instance, achieving a particular security level may necessitate a ring dimension significantly smaller than the next-largest power of two. Restricting parameters to powers of two can thus result in key sizes and runtimes that are at least twice as large as necessary, impacting practicality. This flexibility is essential for developing implementations that meet targeted performance benchmarks or operational constraints. Powers of two are indeed inherently sparsely distributed, as illustrated in the table below, where we have enumerated all possible values of $M = t^\alpha$ ranging from 500 to 10000, with those corresponding to $t = 2$ highlighted in **bold**.

t	α	t^α	t	α	t^α	t	α	t^α	t	α	t^α
2	9	**512**	37	2	1369	53	2	2809	73	2	5329
23	2	529	41	2	1681	5	5	3125	79	2	6241
5	4	625	43	2	1849	59	2	3481	3	8	6561
3	6	729	2	11	**2048**	61	2	3721	19	3	6859
29	2	841	3	7	2187	2	12	**4096**	83	2	6889
31	2	961	13	3	2197	67	2	4489	89	2	7921
2	10	**1024**	47	2	2209	17	3	4913	2	13	**8192**
11	3	1331	7	4	2401	71	2	5041	97	2	9409

  In Section 7.5, a plot demonstrates that using non-power-of-two values for N provides greater flexibility in supporting a fixed number of additions within the cryptosystem. Of course, allowing for general indices M further broadens the range of options available.

- **Flexibility in Using NTT-like Algorithms:** The most computationally intensive step during bootstrapping is performing a series of multiplications within the cyclotomic ring $\mathbb{Z}_q[X]/\Phi_M(X)$. In most current implementations, both q and N are chosen as powers of two-commonly $q = 2^{32}$ or $q = 2^{64}$, with $2^{10} \leq N \leq 2^{16}$. Due to the large polynomial degrees, Fast Fourier Transform (FFT) techniques are employed to perform these multiplications efficiently. However, FFT methods introduce approximation errors, which can impact the accuracy of the computations. The approach proposed here offers greater flexibility in utilizing NTT-like algorithms such as SchÃ¶nhage-Strassen or Nussbaumer, providing an alternative to FFT-based methods, especially when FFT is ill-conditioned. Indeed, more options exist for selecting NTT-friendly parameters, such as choosing $q \bmod M = 1$ or just $\gcd(q, M) = 1$. This enhanced adaptability can lead to more robust and efficient implementations of FHE.

- **Beginning the Transition to General Cyclotomic Polynomials:** In addition to providing more options for parameter selection (as illustrated in the second point regarding $M = t^\alpha$), advancing towards fully general cyclotomic polynomials (which will be explored in a forthcoming publication) offers several theoretical and computational benefits. Certain applications require the use of non-power-of-two cyclotomic rings because power-of-two cyclotomic rings often lack the algebraic properties necessary for efficient SIMD operations or for managing plaintext spaces isomorphic to finite fields other than $\mathbb{Z}_2$ itself. An important additional motivation is the diversification of security assumptions (a point highlighted in [30]): while it is conjectured that ring-LWE remains hard across all high-order cyclotomic rings, some of these rings may be significantly more vulnerable than others.

- **Potential for Optimized Bootstrapping:** With access to a wider variety of polynomial structures, it becomes possible to develop strategies that improve both the efficiency and effectiveness of noise mitigation techniques. In particular, a novel approach-bootstrapping the error rather than the message-is presently being studied and will be detailed in a forthcoming publication.

The standard TFHE [16] and FHEW [21] schemes, primarily operating under the assumption that $M = 2^k$, have rarely been practically extended to more general cyclotomic rings. In fact, such extensions have typically been limited to cyclotomic polynomials of the forms $M = 2^\alpha 3^\beta$ [24, 25] or $M = t^\alpha$ as shown in [4], particularly for homomorphically testing the equality of an encrypted message with a specified plaintext message. Here, we propose an extension of both the FHEW and TFHE *comprehensive* frameworks to include the class of M-th cyclotomic polynomials with $M = t^\alpha$, where t is any prime integer greater than or equal to 3, and α is any non-zero integer. More concretely, whereas traditional TFHE systems often operate within the polynomial ciphertext space defined as:

$$\mathbb{Z}[X]/(X^{2^k}+1) \quad \text{and} \quad \mathbb{T}[X]/(X^{2^k}+1), \quad \mathbb{T} = \mathbb{Q}/\mathbb{Z},$$

we are addressing a broader scenario characterized by:

$$\mathcal{R} := \mathbb{Z}[X]/(\Phi_M(X)) \quad \text{and} \quad \mathcal{T} := \mathbb{T}[X]/(\Phi_M(X)),$$

where M adopts the aforementioned form $M = t^\alpha$. The extension to the fully general case $M = t_1^{\alpha_1} t_2^{\alpha_2} \cdots t_r^{\alpha_r}$, which introduces additional complexities as well as new possibilities, will be elaborated upon in Part II.

In the upcoming sections, we will explore the processes of extraction, bootstrapping, and efficient homomorphic evaluation of the trace operator. We will investigate tasks such as *blind extraction* and *fast packing*, and the algorithms we develop are expected to contribute innovative solutions in this context. Our aim is to clarify the objectives of each procedure and demonstrate how existing methods can be adapted, enhanced, or streamlined for our current applications. Before we delve into these topics, Section 3 will present the essential algebraic tools necessary for constructing the various algorithms discussed. Furthermore, we will offer a brief overview of standard procedures for LWE and RLWE, including encryption, decryption, addition, and external product, as described in Section 5. It is essential to highlight that throughout this text, the scalar messages considered are contained within $\frac{1}{p}\mathbb{Z}_p$ for moderate values of p, with applications to large integers via the Residual Number System in mind, as discussed for instance in [8, 9, 10, 11].

We will now summarize the main content of the paper: Section 6 centers on the extraction of a coefficient μ_i from the encrypted polynomial $\mu(X)$. While this process is

standard practice, we introduce a systematic framework for defining it within the context of any prime-power cyclotomic field. This section also emphasizes the significance of the dual basis related to the scalar product, which we express in terms of the trace operator (refer to Section 3). Notably, the application of duality techniques facilitates a natural approach to scenarios where the index i is also encrypted.

In Section 7, we present a new formulation of the standard functional bootstrapping procedure in terms of the trace operator and the dual basis. We show that this formulation allows for a clear articulation of the "negacyclic" conditions that the functions to be bootstrapped to must satisfy, particularly within the context of prime-power cyclotomic fields. Furthermore, we demonstrate that employing non-power-of-two cyclotomic rings does not adversely affect the error performance. Specifically, we provide a noise analysis of the bootstrap output for general $t \geq 3$, showing that the increase in variance when transitioning from $t = 2$ to $t \geq 3$ is bounded by a factor of two-precisely the growth in variance associated with a single addition. Note that our analysis recovers some results from [20]. It is important to note that we have postponed a complete security analysis to the upcoming Part II. FHE schemes are indeed characterized by two primary security concepts, the latest of which, IND-CPA^D, recently introduced in [29], accounts for potential decryption failures. In the presence of a negligible failure probability, achieving IND-CPA^D security often necessitates selecting parameters that are significantly larger than those sufficient for IND-CPA security. One primary source of such failures is the modulo-switch operation, which introduces a drift in noise levels [3]. An in-depth analysis of this aspect within our broader framework-particularly in the setting of fully general cyclotomic polynomials-will be addressed in future work.

Section 8 provides a comprehensive overview of the various trace operations discussed throughout this paper. Building on the strategy introduced by [13] for power-of-two cyclotomic fields, we will demonstrate how classical Galois theory can be effectively employed within our specific algebraic framework to improve the efficiency of computing the trace from a field to its subfield. This constitutes a nontrivial extension to the general case of prime-power cyclotomic indices, and we believe that the trace operator offers significant advantages for a range of applications.

Finally, Section 9 will explore various variants of the packing operation, which combines multiple LWE encryptions into a single polynomial encryption. The structure of the fields considered in this paper introduces additional complexities to standard algorithms. Once again, we will revisit acceleration techniques, as demonstrated in [13], and show how they can be effectively adapted while maintaining computational efficiency.

To conclude this introductory section, we would like to note that a Part II will soon be submitted for publication, building upon this work by extending all the techniques and procedures discussed here to composite cyclotomic rings. As a specific application, we will demonstrate how large collections of encrypted data can be efficiently managed using encrypted coordinates. Additionally, we will provide error estimates for all operations and propose practical parameter choices.

2 Related works

The selection of underlying algebraic structures, particularly cyclotomic polynomials, is a crucial element of Fully Homomorphic Encryption (FHE) schemes, as these choices significantly influence efficiency and functionality. Notably, several studies have investigated the adaptation of the Number Theoretic Transform (NTT) for these polynomials, which can greatly enhance the speed of polynomial multiplication, with efficiency gains being

especially pronounced for specific parameter selections, as demonstrated by Bajard et al. [2] in their RNS variant of FV-like schemes. In addition to efficiency, noise growth behavior presents another critical concern, particularly as it differs from what is observed in power-of-two cyclotomic rings; this issue was examined in the recent paper by De Micheli et al. [20], underscoring the importance of understanding noise dynamics for establishing the parameters and overall effectiveness of FHE schemes. For instance, Kim et al. [26] explore how these polynomials specifically impact noise growth within the context of private query processing. When employing prime-power cyclotomic polynomials, selecting optimal parameters becomes increasingly complex due to the interplay between polynomial structure, noise growth, and security considerations. Costache and Smart [18] provide a comprehensive comparison of various ring choices, offering detailed evaluations of schemes that utilize different prime-power cyclotomic structures. Given the critical nature of security in FHE schemes based on prime-power cyclotomic polynomials, Eric Crockett and Chris Peikert [19] explore significant obstacles associated with the Ring Learning With Errors (Ring-LWE) problem. They discuss various factors that complicate the implementation and security of RLWE-based schemes, including issues related to parameter selection, the implications of error distributions, and the impact of specific polynomial structures on both security and efficiency. Finally, extension to multivariate RLWE with several cyclotomic polynomials is for instance considered in [5] while Chen et al. [14] investigate potential vulnerabilities and attack strategies that could be exploited due to the unique characteristics of Galois non-dual RLWE families.

3 Algebraic setting

In this section, we present several basic definitions essential for the foundation of the encryption protocol. While some of this material could have been introduced later in the paper, doing so would have compromised clarity, as many concepts here also possess more compact though more abstract definitions we could have relied upon. The identification of the dual sets $\mathcal{R}^\vee$ and $\mathcal{T}^\vee$ with their expressions is provided below.

3.1 Prime power cyclotomic polynomials

We begin by briefly revisiting the definition of cyclotomic polynomials, along with some of their properties and representations when M is a power of a prime.

Definition 3.1 *The M^{th} cyclotomic polynomial is defined by the formula*

$$\Phi_M(X) = \prod_{\substack{1 \le k \le M \\ \gcd(k,M)=1}} \left(X - e^{2i\pi\frac{k}{M}}\right).$$

The cyclotomic polynomials are monic polynomials with integer coefficients that are irreducible over the field of the rational numbers. Except for $M = 1, 2$, they are palindromes of even degree. The degree of Φ_M, or in other words the number of M^{th}- primitive roots of unity, is $N = \varphi(M)$, where φ is Euler's totient function. As we consider indices of the form $M = t^\alpha$, where t is a prime number and $\alpha \ge 1$, only two relational definitions are necessary for computing the cyclotomic polynomials we need:

1. Cyclotomic Polynomial for a Prime: If t is a prime number, the t^{th} cyclotomic

polynomial $\Phi_t(X)$ is defined as:

$$\Phi_t(X) = 1 + X + X^2 + \cdots + X^{t-1} = \sum_{k=0}^{t-1} X^k.$$

This polynomial has roots that are the primitive t^{th} roots of unity, and it is irreducible over the integers.

2. Cyclotomic Polynomial for Prime Powers: If $M = t^\alpha$ with $\alpha \geq 2$, where t is prime, the M^{th} cyclotomic polynomial can be expressed as:

$$\Phi_{t^\alpha}(X) = \Phi_t(X^{t^{\alpha-1}}) = \sum_{k=0}^{t-1} X^{kt^{\alpha-1}}.$$

This relation indicates that the cyclotomic polynomial for the prime power t^α is derived by substituting $X^{t^{\alpha-1}}$ into the t^{th} cyclotomic polynomial. The resulting polynomial captures the multiples of $t^{\alpha-1}$ in its exponents.

We end up this subsection by stating a useful (and well-known) relation on cyclotomic polynomials:

Proposition 3.2 *Let M be a positive integer and d a coprime with M. Then*

$$\Phi_M(X) | \Phi_M(X^d).$$

3.2 The cyclotomic field $\mathcal{K}$ and the ring of algebraic integers $\mathcal{R}$

All sets of polynomials examined in the following sections are embedded within the set of polynomials with rational coefficients modulo Φ_M, which we will refer to as $\mathcal{K}$. This set constitutes a Galois extension of $\mathbb{Q}$ and possesses the structure of both a field and a $\mathbb{Q}$-linear space with dimension $N = \varphi(M)$. It is thus natural to consider the action of the trace operator on $\mathcal{K}$, the associated scalar product on $\mathcal{K}^2$ and to develop the dual basis of the canonical basis from which both $\mathcal{R}^\vee$ and $\mathcal{T}^\vee$ can be described. In this paper, we will use the notation $P \bmod \Phi_M$ interchangeably to refer either to the equivalence class $P(X) + \Phi_M(X)Q(X)$, where $Q \in \mathbb{Q}[X]$, or to the representative element P of this class.

Definition 3.3 (Field of rational polynomials modulo Φ_M) *The field of polynomials with coefficients in $\mathbb{Q}$ modulo $\Phi_M(X)$ is defined as the quotient ring*

$$\mathcal{K} = \frac{\mathbb{Q}[X]}{(\Phi(X))} = \Big\{ P \bmod \Phi_M,\ P \in \mathbb{Q}_{N-1}[X] \Big\}.$$

The field $\mathcal{K}$ is also a Galois extension of $\mathbb{Q}$ of dimension $N = \varphi(M)$. We denote the ring of algebraic integers of $\mathcal{K}$ as $\mathcal{R}$: it is made of the equivalence classes of $\mathcal{K}$ which have representatives in $\mathbb{Z}_{N-1}[X]$, that is to say

$$\mathcal{R} = \Big\{ P \bmod \Phi_M,\ P \in \mathbb{Z}_{N-1}[X] \Big\}.$$

The trace operator in the context of Galois extensions is a linear map from a field extension (here $\mathcal{K}$) back to its base field (here $\mathbb{Q}$). Denoting the associated Galois group $\mathrm{Gal}(\mathcal{K}/\mathbb{Q})$, the trace of an element $P \in \mathcal{K}$ is given by

$$\mathrm{Tr}_{\mathcal{K}/\mathbb{Q}}(P) = \sum_{\tau \in \mathrm{Gal}(\mathcal{K}/\mathbb{Q})} \tau(P) \in \mathbb{Q},$$

where the sum accounts for the action of each automorphism τ in the Galois group (that is to say here the group of substitutions $X \mapsto X^d$ for all $1 \leq d \leq M$ coprime with M) on P. As the trace plays a crucial role in studying the structure and properties of $\mathcal{K}$, we give below its definition instantiated in the context of cyclotomic rings:

Definition 3.4 (Trace operator) *The trace operator* Tr *is defined as the linear map*

$$\begin{aligned} Tr: \quad \mathcal{K} &\longrightarrow \mathbb{Q} \\ P(X) &\longmapsto \sum_{\gcd(d,M)=1} P(X^d) \bmod \Phi_M \end{aligned} \tag{1}$$

The trace is a well-defined operator on $\mathcal{K}$, as its value does not depend on the particular representative of a class in $\mathcal{K}$: for any d such that $\gcd(d, M) = 1$, the cyclotomic polynomial $\Phi_M(X)$ indeed divides $\Phi_M(X^d)$ (Proposition 3.2), so that

$$\sum_{\gcd(d,M)=1} (P + k\Phi_M)(X^d) = \sum_{\gcd(d,M)=1} P(X^d) \bmod \Phi_M.$$

We are now in position to introduce the following scalar product:

Definition 3.5 (Scalar product) *The scalar product* $\langle \cdot, \cdot \rangle$ *is defined on* $\mathcal{K}^2$ *as follows*

$$\forall (P, Q) \in \mathcal{K}^2, \quad \langle P, Q \rangle = \mathrm{Tr}\big(P(X)\bar{Q}(X)\big) \in \mathbb{Q} \tag{2}$$

with $\bar{Q}(X) = Q(X^{-1}) \bmod \Phi_M = Q(X^{M-1}) \bmod \Phi_M$.

Once more, this definition demonstrates consistency since the value of $\langle P, Q \rangle$ remains invariant regardless of the particular representatives selected for $P \bmod \Phi_M$ and $Q \bmod \Phi_M$. This invariance is a consequence of the relation $\bar{\Phi}_M = 0 \bmod \Phi_M$ (see Proposition 3.2). Furthermore, it is straightforward to verify that this scalar product adheres to the typical requirements found in vector spaces, such as bilinearity, symmetry, and positive definiteness. This ensures that $\langle P, Q \rangle$ is both a well-defined and fundamentally sound operation within the context of this algebraic framework.

Now, given any basis $(B_i)_{0 \leq i < N}$ of $\mathcal{K}$, it possesses a dual basis that we shall denote $(B_i^*)_{0 \leq i < N} \subset \mathcal{K}$ defined by

$$\forall 0 \leq i, j < N, \quad \langle B_i^*, B_j \rangle = \delta_{i,j}.$$

It is important to notice that the dual basis $(\Omega_i^*)_{0 \leq i < N}$ of the canonical basis $(X^i)_{0 \leq i < N}$ has the following explicit form (see Proposition 8.1)

$$\begin{aligned} \Omega_i^*(X) &= \frac{1}{M}\left(X^i - X^{N + [i]_{M/t}}\right), \quad 0 \leq i < N, \\ &= \frac{1}{M}\left(\Omega_i(X) + \sum_{\substack{0 \leq j \leq N-1 \\ \text{s.t. } [j-i]_{\frac{M}{t}} = 0}} \Omega_j(X)\right) \end{aligned} \tag{3}$$

where for $0 \leq i < N$, $[i]_{M/t}$ denotes the integer $0 \leq j < \frac{M}{t}$ such that $i = j \bmod \frac{M}{t}$. Conversely, the inverse formula is given by

$$\Omega_i(X) = M\Omega_i^*(X) - \frac{M}{t} \sum_{\substack{0 \leq j \leq N-1 \\ \text{s.t. } [j-i]_{\frac{M}{t}} = 0}} \Omega_j^*(X) \tag{4}$$

A few remarks are now in order. First of all, since $\Omega_0^*(X) = \frac{1-X^N}{M}$ and $\Phi_M(X)$ are coprime elements of the euclidean ring $\mathbb{Q}[X]$, the Bezout identity asserts that there exist U and V in $\mathbb{Q}[X]$ such that

$$\Omega_0^*(X)U(X) + \Phi_M(X)V(X) = 1$$

so that

$$U = (\Omega_0^*)^{-1} \bmod \Phi_M$$

and it can be checked that whereas $V(X) = 1 + \frac{1}{t}\left(X^N - \Phi_M(X)\right) \in \mathcal{T}$,

$$(\Omega_0^*)^{-1}(X) = \underbrace{t^{\alpha-1}\Big(1 + \sum_{k=1}^{t-2} (k+1-t)\, X^{k\frac{M}{t}}\Big)}_{\in \mathbb{Z}_{N-1}[X]} \bmod \Phi_M$$

is the representative of an element of $\mathcal{R}$. In particular, it has integer coefficients. This implies, not only that the N-dimensional $\mathbb{Q}$-linear space

$$\mathcal{K} = \Big\{ \sum_{i=0}^{N-1} \lambda_i X^i \bmod \Phi_M,\ (\lambda_0, \ldots, \lambda_{N-1}) \in \mathbb{Q}^N \Big\}$$

coincides, through the paring $\langle \cdot, \cdot \rangle$ and the usual identification of linear forms with elements of $\mathcal{K}$, with its dual space

$$\mathcal{K}^\vee = \Big\{ \sum_{i=0}^{N-1} \omega_i^* \Omega_i^* \bmod \Phi_M,\ (\omega_0^*, \ldots, \omega_{N-1}^*) \in \mathbb{Q}^N \Big\} = \Omega_0^* \mathcal{K} = \mathcal{K}$$

where the last equality stems from the fact that $\mathcal{K}$ is a field, but also that the dual $\mathcal{R}^\vee$ of $\mathcal{R}$ can be identified similarly to $\Omega_0^*\mathcal{R}$. As a matter of fact, a linear form ℓ on

$$\mathcal{R} = \Big\{ \sum_{i=0}^{N-1} n_i X^i \bmod \Phi_M,\ (n_0, \ldots, n_{N-1}) \in \mathbb{Z}^N \Big\}$$

is the restriction to $\mathcal{R}$ of a linear form defined on $\mathcal{K}$ and is thus the action though the pairing $\langle \cdot, \cdot \rangle$ of an element $\Omega^* \bmod \Phi_M \in \mathcal{K}^\vee = \mathcal{K}$ with $\Omega^* = \sum_{i=0}^{N-1} \omega_i^* \Omega_i^*$, i.e.

$$\exists \Omega^* \bmod \Phi_M \in \mathcal{K}^\vee,\ \forall P \in \mathcal{R},\ \ell(P) = \langle \Omega^*, P \rangle \in \mathbb{Q}.$$

For the image of ℓ to be included in $\mathbb{Z}$, as is required for a linear form on the $\mathbb{Z}$-module $\mathcal{R}$, it can be seen that for all $P \bmod \Phi_M \in \mathcal{R}$ with $P(X) = \sum_{i=0}^{N-1} n_i X^i$, one must have

$$\forall n \in \mathbb{Z}^N, \ell(P) = \sum_{i=0}^{N-1} \omega_i^* n_i \in \mathbb{Z}$$

that is to say $\omega^* \in \mathbb{Z}^N$, or in other words

$$\Omega^* \bmod \Phi_M \in \mathcal{R}^\vee = \Big\{ \sum_{i=0}^{N-1} \omega_i^* \Omega_i^* \bmod \Phi_M,\ (\omega_0^*, \ldots, \omega_{N-1}^*) \in \mathbb{Z}^N \Big\} = \Omega_0^* \mathcal{R}.$$

The last equality is derived from the fact that $(\Omega_0^*\Omega_j)_{j=0}^{N-1}$, which forms a basis for the $\mathbb{Q}$-linear space $\mathcal{K}$, also acts as a basis for $\mathcal{R}^\vee$. Specifically, we have:

$$\langle \Omega_0^*\Omega_j, \Omega_k\rangle = \langle \Omega_0^*, \Omega_{-j}\Omega_k\rangle \in \mathbb{Z}$$

owing to the fact that $\Omega_{-j}\Omega_k \in \mathcal{R}$. In particular, the change of basis

$$\begin{pmatrix} 1 \\ (\Omega_0^*)^{-1}\Omega_1^* \\ \vdots \\ (\Omega_0^*)^{-1}\Omega_{N-1}^* \end{pmatrix} = \Phi_t(J^{\frac{M}{t}}) \begin{pmatrix} 1 \\ X \\ \vdots \\ X^{N-1} \end{pmatrix} = \Phi_M(J) \begin{pmatrix} 1 \\ X \\ \vdots \\ X^{N-1} \end{pmatrix} \bmod \Phi_M$$

where

$$J = \begin{pmatrix} 0 & \dots & \dots & \dots & 0 \\ 1 & \ddots & & & \vdots \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 1 & 0 \end{pmatrix} \in \mathcal{M}_N(\mathbb{Z}),$$

is cleary unimodular with inverse $(I - J^{\frac{M}{t}}) \in \mathcal{M}_N(\mathbb{Z})$. Furthermore, we observe in passing that $\mathcal{R}^\vee \subset \frac{1}{M}\mathcal{R}$.

3.3 The module $\mathcal{T}$ of polynomials with coefficients in $\mathbb{T}$ and its dual $\mathcal{T}^\vee$

We start by recalling that the torus $\mathbb{T}$ is defined as the set of rational numbers modulo 1, i.e. $\mathbb{T} = \mathbb{Q}/\mathbb{Z}$.

Definition 3.6 *The quotient of the $\mathbb{Z}[X]$-module $\mathcal{K}$ by the sub-module $\mathcal{R}$ is the quotient $\mathbb{Z}[X]$-module $\mathcal{T} = \mathcal{K}/\mathcal{R}$, that is to say the set of equivalence classes*

$$\mathcal{T} = \Big\{ \sum_{i=0}^{N-1} x_i X^i \ mod\ \mathbb{Z}[X]\ mod\ \Phi_M, (x_0, \dots, x_{N-1}) \in \mathbb{Q}^N \Big\}.$$

Definition 3.7 *The quotient of the $\mathbb{Z}[X]$-module $\mathcal{K}$ by the sub-module $\mathcal{R}^\vee$ is the quotient $\mathbb{Z}[X]$-module $\mathcal{T}^\vee = \mathcal{K}/\mathcal{R}^\vee$, that is to say the set of equivalence classes*

$$\mathcal{T}^\vee = \Big\{ \sum_{i=0}^{N-1} \omega_i^*\Omega_i^*(X)\ mod\ \Omega_0^*\mathbb{Z}[X]\ mod\ \Phi_M, (\omega_0^*, \dots, \omega_{N-1}^*) \in \mathbb{Q}^N \Big\}.$$

Remark 3.8 *The polynomial product is well-defined from $\mathcal{R}^\vee \times \mathcal{T}$ to $\mathcal{T}^\vee$, as well as from $\mathcal{R} \times \mathcal{T}^\vee$ to $\mathcal{T}^\vee$. Specifically, we have:*

$$\begin{aligned} \Big(\Omega_0^*(X)Z_1(X) + \Phi_M(X)Q_1(X)\Big) \times \Big(P_2(X) + Z_2(X) + \Phi_M(X)Q_2(X)\Big) \\ = \Omega_0^*(X)Z_1(X)P_2(X) + \Omega_0^*(X)Z_1(X)Z_2(X) + \Phi_M(X)Q_3(X) \\ = \Omega_0^*(X)Z_1(X)P_2(X)\ mod\ \mathcal{R}^\vee \end{aligned}$$

And symmetrically,

$$\begin{aligned} \Big(Z_1(X) + \Phi_M(X)Q_1(X)\Big) \times \Big(\Omega_0^*(X)P_2(X) + \Omega_0^*(X)Z_2(X) + \Phi_M(X)Q_2(X)\Big) \\ = \Omega_0^*(X)Z_1(X)P_2(X) + \Omega_0^*(X)Z_1(X)Z_2(X) + \Phi_M(X)Q_3(X) \\ = \Omega_0^*(X)Z_1(X)P_2(X)\ mod\ \mathcal{R}^\vee \end{aligned}$$

Here, Z_1 and Z_2 are polynomials in $\mathbb{Z}[X]$, while P_2, Q_1, Q_2, and Q_3 are polynomials in $\mathbb{Q}[X]$. Note that the multiplication of a polynomial of $\mathcal{R}$ by a a polynomial of $\mathcal{T}$ also makes sense, in contrast with the case where both are in the duals $\mathcal{R}^\vee$ and $\mathcal{T}^\vee$.

Remark 3.9 *Consider a linear form $\ell \in \mathcal{L}(K, \mathbb{Q})$ represented by*

$$\sum_{i=0}^{N-1} \omega_i^* \Omega_i^*(X) + \Phi_M(X)\tilde{Q}(X)$$

in $\mathcal{K}^\vee = \mathcal{K}$, where $\tilde{Q}$ is a polynomial in $\mathbb{Q}[X]$. The application ℓ can be consistently defined on $\mathcal{T}$ as follows:

$$\begin{array}{rcl} \ell : \mathcal{T} & \to & \mathbb{T} \\ \sum_{i=0}^{N-1} x_i X^i + Z(X) + \Phi_M(X)Q(X) & \mapsto & \Big\langle \sum_{i=0}^{N-1} \omega_i^* \Omega_i^*(X), \sum_{i=0}^{N-1} x_i X^i \Big\rangle \ mod\ 1 \end{array}$$

provided that $(\omega_0^*, \ldots, \omega_{N-1}^*) \in \mathbb{Z}^{N-1}$. In fact, the value of

$$\begin{aligned} &\Big\langle \sum_{i=0}^{N-1} \omega_i^* \Omega_i^*(X) + \Phi_M(X)\tilde{Q}(X), \sum_{i=0}^{N-1} x_i X^i + Z(X) + \Phi_M(X)Q(X) \Big\rangle \\ &\qquad = \sum_{i=0}^{N-1} \omega_i^* x_i + \Big\langle \sum_{i=0}^{N-1} \omega_i^* \Omega_i^*(X), Z(X) \Big\rangle \ mod\ 1 \end{aligned}$$

depends on Z unless $\langle \sum_{i=0}^{N-1} \omega_i^* \Omega_i^*, Z \rangle \in \mathbb{Z}$, which necessitates that $(\omega_0^*, \ldots, \omega_{N-1}^*) \in \mathbb{Z}^{N-1}$. In other words, ℓ is defined by an element of $\mathcal{R}^\vee$.

4 Plaintext messages in the TFHE framework

In the context of TFHE, all messages, also referred to as *plaintext messages* or simply *plaintexts*, can be classified into two types: either as elements of the torus $\mathbb{T}$, or as polynomials within a cyclotomic ring, with coefficients that belong either to $\mathbb{T}$ or $\mathbb{Z}$.

4.1 The set of torus plaintexts

Despite the fact that all elements of the torus can be encrypted, only a discretized subset can be safely decrypted. This observation leads to the following definition:

Definition 4.1 (Discretized torus for messages) *Let $p \geq 3$ be an odd integer. The structure of the discrete torus $\mathbb{T}_p$ is inherited from $(\mathbb{Z}_p, +, \times)$, with privileged representative*[1]

$$i \ mods\ p.$$

The discrete torus $\mathbb{T}_p \subset \mathbb{T} = [-\frac{1}{2}, \frac{1}{2}) + \mathbb{Z}$ is defined by $\mathbb{T}_p = \frac{1}{p}\mathbb{Z}_p$:

$$\mathbb{T}_p = \big\{ -(p-1)/(2p), ..., (p-1)/(2p) \big\} + \mathbb{Z}$$

[1] The symmetric modulo operation returns the remainder of a division such that the result is centered around zero. For a number i and odd modulus p, it maps i to the interval $\big\{ -(p-1)/2, ..., (p-1)/2 \big\}$, ensuring the result is balanced symmetrically around zero.

This definition is crucial in the design of schemes such as TFHE, where ensuring the integrity of decrypted messages relies on restricting the plaintext space to a manageable and well-defined subset. The choice of this discretized subset ultimately influences the efficiency, security, and overall functionality of the homomorphic encryption system. It allows the framework to balance between operational flexibility and the necessary constraints imposed by noise growth during cryptographic operations. Note that $\mathbb{T}_p$, the p-adic torus, is a ring that is isomorphic to $(\mathbb{Z}_p, +, \times)$. Its structure can be outlined as follows:

1. Addition: The addition operation in $\mathbb{T}_p$ is inherited from the torus $\mathbb{T}$. Specifically, for any $(x, y) \in \mathbb{T}_p \times \mathbb{T}_p$:
$$x + y \equiv x + y \bmod 1.$$

2. Multiplication: The multiplication in $\mathbb{T}_p$ is inherited from the integers modulo p. For any $(x, y) \in \mathbb{T}_p \times \mathbb{T}_p$:
$$x \times y = (px) \times y \bmod 1.$$

4.2 The set of polynomial plaintexts

The extension to prime power cyclotomic rings manifests itself when considering polynomial messages and all subsequent attached procedures (packing, extraction, bootstrapping...).

Definition 4.2 (Polynomial plaintext) *Let M be a non-zero integer and let Φ_M be the M^{th} cyclotomic polynomial. Polynomial plaintexts are polynomial representatives of equivalence classes, either in $\mathcal{R}$ or in*

$$\mathcal{T}_p = \left\{ P(X) \ mod \ \mathbb{Z}[X] \ mod \ \Phi_M, P \in \frac{1}{p}\mathbb{Z}[X] \right\} \subset \mathcal{T}.$$

The polynomials in $\mathcal{T}_p$ can not only be encrypted and manipulated but also decrypted exactly with a high probability under standard parameter conditions (see next Section).

5 LWE and RLWE encryptions cryptographic schemes

5.1 Encryption/decryption schemes in $\mathbb{T}$

Learning With Errors (LWE) is a cryptographic problem widely used in post-quantum cryptography due to its hardness against quantum attacks [34]. The LWE problem consists in solving systems of noisy linear equations. More specifically, given a secret vector $\mathbf{s} \in \mathbb{S}^n$, where $\mathbb{S}$ is a finite subset of $\mathbb{Z}$, if we are given a set of linear equations of the form

$$\mathbf{c} = \mathbf{A} \cdot \mathbf{s} + \mathbf{e},$$

where $\mathbf{A}$ is randomly chosen matrix over a finite field (e.g., modulo $\mathbb{Z}_q$) and $\mathbf{e}$ is a random noise vector with small entries, recovering $\mathbf{s}$ from the equations is computationally hard. Of course, if $\mathbf{A} \in \mathcal{M}_n(\mathbb{Z}_q)$ and $\mathbf{e} \in (\mathbb{Z}_q)^n$, the actual hardness depends on how large q and n are. When it is hard enough, the LWE-problem is said to be secure.

Now, assume the LWE-problem on $\mathbb{Z}_q$ is secure and assimilate $\mathbb{T} \equiv \frac{1}{q}\mathbb{Z}_q$ for the purpose of the following definition [15, 16].

Definition 5.1 (EncryptLWE$_{\mathbf{s}}(\mu)$) *The* LWE*-encryption of a message $\mu \in \mathbb{T}$ with the secret key $\mathbf{s} \in \mathbb{S}^n$ is defined as*

$$\mathbf{c} = \mathrm{LWE}_{\mathbf{s}}(\mu) = (\mathbf{a}, b) \in \mathbb{T}^{n+1}$$

with

$$\mathbf{a} = (a_1, \ldots, a_n) \xleftarrow{\$} \mathbb{T}^n, \quad e \leftarrow \mathcal{N}(0, \sigma^2)$$

and

$$b = \mathbf{s} \cdot \mathbf{a} + \mu + e.$$

Algorithm 1 LWE Encryption of Message μ

Input: Message $\mu \in \mathcal{T}$, secret key $\mathbf{s} \in \mathbb{S}^n$.

1: **Generate random vector:** $\mathbf{a} = (a_1, \ldots, a_n) \xleftarrow{\$} \mathcal{T}^n$
2: **Sample error term:** $e \leftarrow \mathcal{N}(0, \sigma^2)$
3: $b \leftarrow \mathbf{s} \cdot \mathbf{a} + \mu + e$
4: **Return** $\mathbf{c} = (\mathbf{a}, b)$.

Definition 5.2 (DecryptLWE$_{\mathbf{s}}$ $(\mathbf{c}, p)$) *The* LWE*-decryption of a ciphertext $(\mathbf{a}, b) \in \mathbb{T}^{n+1}$ with secret key $\mathbf{s} \in \mathbb{S}^n$ is defined as*

$$\pi_p(b - \mathbf{s} \cdot \mathbf{a}) \in \mathbb{T}_p$$

where π_p is a projection on the discrete torus $\mathbb{T}_p$.

Algorithm 2 LWE Decryption

Input: Ciphertext $(\mathbf{a}, b) \in \mathcal{T}^{n+1}$, secret key $\mathbf{s} \in \mathbb{S}^n$, modulus p.

1: $\varphi \leftarrow b - \mathbf{s} \cdot \mathbf{a}$
2: **Return**

$$\frac{\lfloor p\varphi \rceil}{p} \in \mathbb{T}_p.$$

It is clear that if $\mathbf{c} = (\mathbf{a}, b)$ is the LWE-encryption of a message μ in $\mathbb{T}_p$, then

$$\pi_p(b - \mathbf{a} \cdot \mathbf{s}) = \mu$$

if $|e|$ is small enough, more precisely if $|e| < \frac{1}{2p}$. Here, the projection π_p is defined as $\pi_p(\mu) = \frac{\lfloor p\mu \rceil}{p}$ for all $\mu \in \mathbb{T}$.

5.2 Encryption/decryption schemes in $\mathcal{T}$ and $\mathcal{T}^\vee$

The security of Ring Learning With Errors (Ring-LWE) is rooted in the same principles as the standard LWE problem [35]. As mentioned earlier, the dual sets $\mathcal{T}$ and $\mathcal{T}^\vee$ respectively, while abstractly well-defined, can be represented in the following respective forms that are convenient to work with:

$$\mathcal{T} = \Big\{ P \bmod \mathbb{Z}[X] \bmod \Phi_M,\ P \in \mathbb{Q}_{N-1}[X] \Big\} = \mathcal{K}/\mathcal{R},$$
$$\mathcal{T}^\vee = \Big\{ \Omega_0^* P \bmod \Omega_0^* \mathbb{Z}[X] \bmod \Phi_M,\ P \in \mathbb{Q}_{N-1}[X] \Big\} = \mathcal{K}/\mathcal{R}^\vee.$$

The Ring-LWE problem (adapted to the torus in this context) as articulated by Regev in [35] is defined as follows:

Ring-LWE problem. For $M > 1$, consider the cyclotomic ring $\mathcal{R}$ and the cyclotomic $\mathbb{Z}[X]$-module $\mathcal{T}^\vee$. Given samples of the form

$$(a^*, b^*) \in \mathcal{T}^\vee \times \mathcal{T}^\vee,$$

with $b^* = s \cdot a^* + e^*$ and where

- a^* is chosen uniformly at random from $\mathcal{T}^\vee$;
- s is a secret element in $\mathcal{R}$ with coefficients in $\mathbb{S}$;
- e^* is a small error term sampled from a normal distribution over $\mathcal{T}^\vee$;

it is a computationally hard problem to distinguish such samples from uniformly random pairs in $\mathcal{T}^\vee \times \mathcal{T}^\vee$.

The security of Ring-LWE is built upon its reduction to hard problems on ideal lattices [35], such as the *Ideal Shortest Vector Problem (Ideal-SVP)* and the *Ideal Closest Vector Problem (Ideal-CVP)*. Once again, the effective level of security depends in particular on how large parameters M and q in $\mathbb{T} \equiv \frac{1}{q}\mathbb{Z}_q$ are.

Now, assume the RLWE-problem is secure. We may encrypt and decrypt polynomial messages as follows:

Definition 5.3 (EncryptRLWE*$_s(\mu^*)$) *The* RLWE**-encryption of a message* $\mu^*(X) \in \mathcal{T}^\vee$ *with the secret key*

$$s(X) = \sum_{i=0}^{N-1} s_i X^i \in \mathcal{R}, \qquad (s_0, \ldots, s_{N-1}) \in \mathbb{S}^N,$$

is defined as

$$c^*(X) = \mathrm{RLWE}^*_s(\mu^*(X)) = (a^*(X), b^*(X)) \in \mathcal{T}^\vee \times \mathcal{T}^\vee$$

with

$$(a^*_0, \ldots, a^*_{N-1}) \xleftarrow{\$} \mathbb{T}^N, \qquad a^*(X) = \sum_{i=0}^{N-1} a^*_i \Omega^*_i(X),$$

$$(e^*_0, \ldots, e^*_{N-1}) \xleftarrow{\mathcal{N}(0,\sigma^2)} \mathbb{T}^N, \qquad e^*(X) = \sum_{i=0}^{N-1} e^*_i \Omega^*_i(X),$$

and

$$b^* = s \cdot a^* + \mu^* + e^* \ mod\ \mathcal{R}^\vee.$$

In other words, the equality holds modulo $\Omega^*_0\mathbb{Z}[X]$ *and* Φ_M*, and* $b^* \in \mathcal{T}^\vee$*.*

Algorithm 3 RLWE* Encryption of a polynomial message in $\mathcal{T}^\vee$

Input: Message $\mu^*(X) \in \mathcal{T}^\vee$ and secret key $s(X) \in \mathcal{R}$.

1: **Generate:** $(a^*_0, \ldots, a^*_{N-1}) \xleftarrow{\$} \mathbb{T}^N$ and $(e^*_0, \ldots, e^*_{N-1}) \xleftarrow{\mathcal{N}(0,\sigma^2)} \mathbb{T}^N$
2: $a^*(X) \leftarrow \sum_{i=0}^{N-1} a^*_i \Omega^*_i(X)$ and $e^*(X) \leftarrow \sum_{i=0}^{N-1} e^*_i \Omega^*_i(X)$
3: $b^*(X) \leftarrow s(X) \cdot a^*(X) + \mu^*(X) + e^*(X) \bmod \mathcal{R}^\vee$.
4: **Return** $c^*(X) = (a^*(X), b^*(X)) \in \mathcal{T}^\vee \times \mathcal{T}^\vee$.

Remark 5.4 *It is also possible to choose $s \in \mathcal{R}^\vee$ and sample $a \in \mathcal{T}$, so that $b \in \mathcal{T}^\vee$ as in [36, 35].*

Definition 5.5 (EncryptRLWE$_s(\mu)$) *The* RLWE*-encryption of a message $\mu(X) \in \mathcal{T}$ with the secret key*

$$s(X) = \sum_{i=0}^{N-1} s_i X^i \in \mathcal{R}, \qquad (s_0, \ldots, s_{N-1}) \in \mathbb{S}^N,$$

is defined as

$$c(X) = \mathrm{RLWE}_s(\mu(X)) = (a(X), b(X)) \in \mathcal{T} \times \mathcal{T}$$

with $a = (\Omega_0^*)^{-1} a^*$ mod Φ_M, $b = (\Omega_0^*)^{-1} b^*$ mod Φ_M and where

$$(a^*(X), b^*(X)) = \mathrm{RLWE}_s^*(\Omega_0^*(X)\mu(X)).$$

Note that

$$b = s \cdot a + \mu + e \bmod \mathcal{R},$$

with $e = (\Omega_0^*)^{-1} e^*$. The equality holds modulo $\mathbb{Z}[X]$ and Φ_M, and $b \in \mathcal{T}$.

Algorithm 4 RLWE Encryption of Message μ

Input: Message $\mu(X) \in \mathcal{T}$, secret key $s(X)$.

1: **Compute the RLWE*-encryption:** $(a^*(X), b^*(X)) = \mathrm{RLWE}_s^*(\Omega_0^*(X)\mu(X))$.
2: $a \leftarrow \Big((\Omega_0^*)^{-1} a^* \bmod \Phi_M\Big) \bmod \mathcal{R}$
3: $b \leftarrow \Big((\Omega_0^*)^{-1} b^* \bmod \Phi_M\Big) \bmod \mathcal{R}$
4: **Return** $c(X) = (a(X), b(X)) \in \mathcal{T} \times \mathcal{T}$.

Definition 5.6 (DecryptRLWE$_s^*$ ($\mathbf{c}^*(\mathbf{X}), p$)) *The* RLWE*-decryption of the ciphertext $c^*(X) = (a^*(X), b^*(X)) \in \mathcal{T}^\vee \times \mathcal{T}^\vee$ with the secret key $s \in \mathcal{R}$ is defined as*

$$\pi_p\Big(b^*(X) - s(X)a^*(X)\Big) \in \mathcal{T}_p$$

where π_p is a projection coefficient by coefficient (in the basis $(\Omega_i^*(X))_{0 \le i < N}$) on the discrete torus $\mathbb{T}_p$.

Algorithm 5 RLWE* Decryption

Input: Ciphertext $c^*(X) = (a^*(X), b^*(X)) \in \mathcal{T}^\vee \times \mathcal{T}^\vee$, secret key $s \in \mathcal{R}$, modulus p.

1: $\varphi^*(X) \leftarrow b^*(X) - s(X)a^*(X) \bmod \mathcal{R}^\vee$.
2: **Return**

$$\tilde{\varphi}(X) = \sum_{i=0}^{N-1} \frac{\lfloor p\varphi_i^* \rceil}{p} \Omega_i^*(X) \in \mathcal{T}_p^\vee.$$

Definition 5.7 (DecryptRLWE$_s$ ($\mathbf{c}(\mathbf{X}), p$)) *The* RLWE*-decryption of the ciphertext $c(X) = (a(X), b(X)) \in \mathcal{T} \times \mathcal{T}$ with the secret key $s \in \mathcal{R}$ is defined as*

$$\pi_p\Big(b(X) - s(X)a(X)\Big) \in \mathcal{T}_p$$

where π_p is a projection coefficient by coefficient in the basis $(\Omega_i)_{0 \le i < N}$ on the discrete torus $\mathbb{T}_p$.

Algorithm 6 RLWE Decryption

Input: Ciphertext $c(X) = (a(X), b(X)) \in \mathcal{T} \times \mathcal{T}$, secret key $s \in \mathcal{R}$, modulus p.

1: $\varphi(X) \leftarrow b(X) - s(X)a(X) = \sum_{i=0}^{N-1} \varphi_i X^i \bmod \mathcal{R}$.

2: **Return**

$$\tilde{\varphi}(X) = \sum_{i=0}^{N-1} \frac{\lfloor p\varphi_i \rceil}{p} X^i \in \mathcal{T}_p.$$

It is clear that if $c(X) = (a(X), b(X))$ is the RLWE-encryption of a message $\mu(X)$ in $\mathcal{T}_p$, then

$$\pi_p\Big(b(X) - s(X)a(X)\Big) = \mu(X)$$

if $\|e\|_\infty$ is small enough, more precisely if $\|e\|_\infty < \frac{1}{2p}$. The norm $\|\cdot\|_\infty$ of a polynomial denotes here, as is customary, the maximum of the absolute values of its coefficients in the basis $(\Omega_i)_{0 \le i < N}$.

5.3 Encryption/decryption schemes in $\mathcal{R}$ and $\mathcal{R}^\vee$

In this subsection, we describe the encryption of elements in the dual sets $\mathcal{R}$ and $\mathcal{R}^\vee$:

$$\mathcal{R} = \Big\{P \bmod \Phi_M,\ P \in \mathbb{Z}_{N-1}[X]\Big\} \subset \mathcal{K} \text{ and } \mathcal{R}^\vee = \Big\{\Omega_0^* P \bmod \Phi_M,\ P \in \mathbb{Z}_{N-1}[X]\Big\} \subset \mathcal{K}.$$

Definition 5.8 (EncryptGRLWE*$_s(\mu^*)$) *Given two intergers $D \ge 2$ and $\ell \ge 1$, the* GRLWE*-*encryption of a message*

$$\mu^*(X) = \sum_{i=0}^{N-1} \mu_i^* \Omega_i^*(X) \in \mathcal{R}^\vee$$

with the secret key

$$s(X) = \sum_{i=0}^{N-1} s_i X^i \in \mathcal{R}, \qquad (s_0, \ldots, s_{N-1}) \in \mathbb{S}^N,$$

is defined as

$$C^*(X) = \mathrm{GRLWE}_s^*(\mu^*(X)) = \begin{pmatrix} \mathrm{RLWE}_s^*\left(\frac{\mu^*(X)}{D}\right) \\ \vdots \\ \mathrm{RLWE}_s^*\left(\frac{\mu^*(X)}{D^\ell}\right) \end{pmatrix}.$$

Algorithm 7 GRLWE* Encryption of Message $\mu^* \in \mathcal{R}^\vee$

Input: Message $\mu^*(X) \in \mathcal{R}^\vee$, secret key $s(X)$, integers $D \ge 2$ and $\ell \ge 1$.

1: **Initialize an empty vector**: $C^*(X) \leftarrow \emptyset$.

2: **for** each k from 1 to ℓ **do**

3: $\quad c_k^*(X) \leftarrow \mathrm{RLWE}_s^*\left(\frac{\mu^*(X)}{D^k}\right)$.

4: $\quad$ **Append** $c_k^*(X)$ to $C^*(X)$.

5: **end for**

6: **Return**

$$C^*(X) = \begin{pmatrix} c_1^* \\ \vdots \\ c_\ell^* \end{pmatrix}.$$

Definition 5.9 (EncryptGRLWE$_s(\mu)$) *Given two intergers $D \geq 2$ and $\ell \geq 1$, the* GRLWE*-encryption of a message*

$$\mu(X) = \sum_{i=0}^{N-1} \mu_i X^i \in \mathcal{R}$$

with the secret key

$$s(X) = \sum_{i=0}^{N-1} s_i X^i \in \mathcal{R}, \qquad (s_0, \ldots, s_{N-1}) \in \mathbb{S}^N,$$

is defined as

$$C(X) = \mathrm{GRLWE}_s(\mu(X)) = \begin{pmatrix} \mathrm{RLWE}_s\left(\frac{\mu(X)}{D}\right) \\ \vdots \\ \mathrm{RLWE}_s\left(\frac{\mu(X)}{D^\ell}\right) \end{pmatrix}.$$

Algorithm 8 GRLWE Encryption of Message $\mu \in \mathcal{R}$

Input: Message $\mu(X) \in \mathcal{R}$, secret key $s(X)$, integers $D \geq 2$ and $\ell \geq 1$.

1:
2: **Initialize an empty vector**: $C(X) \leftarrow \emptyset$.
3: **for** each k from 1 to ℓ **do**
4: $\quad c_k \leftarrow \mathrm{RLWE}_s\left(\frac{\mu(X)}{D^k}\right)$.
5: $\quad$ **Append** c_k to C.
6: **end for**
7: **Return**

$$C(X) = \begin{pmatrix} c_1 \\ \vdots \\ c_\ell \end{pmatrix}.$$

Definition 5.10 (EncryptRGSW*$_s(\mu^*)$) *Given two intergers $D \geq 2$ and $\ell \geq 1$, the* RGSW**-encryption of a message $\mu^*(X) \in \mathcal{T}^\vee$ with the secret key*

$$s(X) = \sum_{i=0}^{N-1} s_i X^i \in \mathcal{R}, \qquad (s_0, \ldots, s_{N-1}) \in \mathbb{S}^N,$$

is defined as

$$c^*(X) = \mathrm{RGSW}^*_s(\mu^*(X)) = \begin{pmatrix} \mathrm{GRLWE}^*_s(-s(X)\mu^*(X)) \\ \mathrm{GRLWE}^*_s(\mu^*(X)) \end{pmatrix}.$$

Algorithm 9 RGSW* Encryption of Message μ^*

Input: Message $\mu^*(X) \in \mathcal{T}^\vee$, secret key $s(X)$, integers $D \geq 2$ and $\ell \geq 1$.

1: **Compute the RGSW* encryptions:** $C_1^* = \mathrm{GRLWE}^*_s(-s(X)\mu^*(X))$ and $C_2^* = \mathrm{GRLWE}^*_s(\mu^*(X))$.
2: **Return**

$$C^*(X) = \begin{pmatrix} C_1^* \\ C_2^* \end{pmatrix}.$$

Definition 5.11 (EncryptRGSW$_s(\mu)$) *Given two intergers $D \geq 2$ and $\ell \geq 1$, the* RGSW-*encryption of a message $\mu(X) \in \mathcal{T}$ with the secret key*

$$s(X) = \sum_{i=0}^{N-1} s_i X^i \in \mathcal{R}, \qquad (s_0, \ldots, s_{N-1}) \in \mathbb{S}^N,$$

is defined as

$$C(X) = \mathrm{RGSW}_s(\mu(X)) = \begin{pmatrix} \mathrm{GRLWE}_s(-s(X)\mu(X)) \\ \mathrm{GRLWE}_s(\mu(X)) \end{pmatrix}.$$

Algorithm 10 RGSW Encryption of Message μ

Input: Message $\mu(X) \in \mathcal{T}$, secret key $s(X)$, integers $D \geq 2$ and $\ell \geq 1$.

1: $C_1 \leftarrow \mathrm{GRLWE}_s(-s(X)\mu(X))$
2: $C_2 \leftarrow \mathrm{GRLWE}_s(\mu(X))$.
3: **Return**

$$C(X) = \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}.$$

5.4 Homomorphic addition

We denote unambiguously by $\oplus$ the addition of (R)LWE-ciphertexts and RGSW-ciphertexts, as well as (R)LWE*-ciphertexts and RGSW*-ciphertexts,. We recall that, if c_1 and c_2 are two (R)LWE-ciphertexts, i.e.

$$c_1 = \text{(R)LWE}_s(\mu_1) \text{ and } c_2 = \text{(R)LWE}_s(\mu_2),$$

then

$$c_1 \oplus c_2 = \text{(R)LWE}_s(\mu_1 + \mu_2)$$

where the equality means that

$$\mathrm{Decrypt}\text{(R)LWE}_s(c_1 \oplus c_2, p) = \mu_1 + \mu_2.$$

Similarly, if C_1 and C_2 are two RGSW-ciphertexts, i.e.

$$C_1 = \mathrm{RGSW}_s(m_1) \text{ and } C_2 = \mathrm{RGSW}_s(m_2)$$

then

$$C_1 \oplus C_2 = \mathrm{RGSW}_s(m_1 + m_2)$$

where the equality means that both sides are (possibly different) encryptions of $m_1 + m_2$. Finally, if C_1^* and C_2^* are two RGSW-ciphertexts, i.e.

$$C_1^* = \mathrm{RGSW}_s^*(m_1^*) \text{ and } C_2^* = \mathrm{RGSW}_s^*(m_2^*)$$

then

$$C_1^* \oplus C_2^* = \mathrm{RGSW}_s^*(m_1^* + m_2^*)$$

In all cases, the effective addition is component-wise (with polynomial coefficients).

5.5 Homomorphic modular product

We recall that the $\mathbb{Z}_N[X]$-module $\mathbb{T}_N[X]$ is by definition endowed with a *modular* product $\cdot$ whose counterpart on RGSW-ciphertexts is the co-called *external* product $\boxdot$. Besides, if

$$C^* = \mathrm{RGSW}_s^*(m^*) \in \mathcal{R}^\vee \text{ and } c = \mathrm{RLWE}_s(\mu) \in \mathcal{T},$$

then

$$C^* \boxdot c = \mathrm{RLWE}_s^*(m^* \cdot \mu) \in \mathcal{T}^\vee$$

in the sense that

$$C^* \boxdot c \text{ is an } \mathrm{RLWE}_s\text{-encryption of } m^* \cdot \mu. \tag{5}$$

The effective external product of C^* and c is obtained through the vector-matrix multiplication (with polynomial coefficients)

$$C^* \boxdot c = \mathrm{dec}_{D,\ell}(c)\ C^*$$

where $\mathrm{dec}_{D,\ell}(c) = (\mathrm{dec}_{D,\ell}(a(X)), \mathrm{dec}_{D,\ell}(b(X)))$ and

$$\mathrm{dec}_{D,\ell}(a(X)) = \left(\sum_{r=0}^{N-1} \mathrm{dec}_{D,\ell}(a_r)_1 X^r, \ldots, \sum_{r=0}^{N-1} \mathrm{dec}_{D,\ell}(a_r)_\ell X^r \right),$$
$$\mathrm{dec}_{D,\ell}(b(X)) = \left(\sum_{r=0}^{N-1} \mathrm{dec}_{D,\ell}(b_r)_1 X^r, \ldots, \sum_{r=0}^{N-1} \mathrm{dec}_{D,\ell}(b_r)_\ell X^r \right),$$

for some integers $D \geq 2$ and $\ell \geq 1$. To complete the definition of $\mathrm{dec}_{D,\ell}(c)$ we decompose any $x \in \mathbb{T} \equiv [-\frac{1}{2}, \frac{1}{2})$ as:

$$x = \sum_{g=1}^{\ell} \mathrm{dec}_{D,\ell}(x)_g D^{-g} + \delta(x), \quad \mathrm{dec}_{D,\ell}(x)_t \in \{-D/2, \ldots, D/2\}, \quad |\delta(x)| \leq \frac{D^{-\ell}}{2}.$$

The dual product $C \boxdot c$ and $C \boxdot c^*$ are defined similarly with the only difference that $\mathrm{dec}_{D,\ell}(c^*) = (\mathrm{dec}_{D,\ell}(a^*(X)), \mathrm{dec}_{D,\ell}(b^*(X)))$ where

$$\mathrm{dec}_{D,\ell}(a^*(X)) = \left(\sum_{r=0}^{N-1} \mathrm{dec}_{D,\ell}(a_r^*)_1 \Omega_r^*(X), \ldots, \sum_{r=0}^{N-1} \mathrm{dec}_{D,\ell}(a_r^*)_\ell \Omega_r^*(X) \right),$$
$$\mathrm{dec}_{D,\ell}(b^*(X)) = \left(\sum_{r=0}^{N-1} \mathrm{dec}_{D,\ell}(b_r^*)_1 \Omega_r^*(X), \ldots, \sum_{r=0}^{N-1} \mathrm{dec}_{D,\ell}(b_r^*)_\ell \Omega_r^*(X) \right).$$

Note that, for instance

$$\mathrm{dec}_{D,\ell}(a(X)) = \sum_{g=1}^{\ell} D^{-g} \left(\sum_{r=0}^{N-1} \mathrm{dec}_{D,\ell}(a_r)_g X^r \right) + \delta(a(X))), \quad \delta(a(X)) = \sum_{r=0}^{N-1} \delta(a_r) X^r$$

and

$$\mathrm{dec}_{D,\ell}(a^*(X)) = \sum_{g=1}^{\ell} D^{-g} \left(\sum_{r=0}^{N-1} \mathrm{dec}_{D,\ell}(a_r^*)_g \Omega_r^*(X) \right) + \delta(a^*(X))), \quad \delta(a^*(X)) = \sum_{r=0}^{N-1} \delta(a_r^*) \Omega_r^*(X).$$

We then have the following standard formulas:

Proposition 5.12 *Let C and $c = (a, b)$ be respectively* RGSW_s *and* RLWE_s *encryptions of m and μ. Then the following equality holds for all integers $D \geq 2$ and $\ell \geq 1$:*

$$\mathrm{Err}(C \boxdot c) = dec_{D,\ell}(c) \cdot \mathrm{Err}(C) + m(\delta(b) - s\ \delta(a)) + m\,\mathrm{Err}(c).$$

Furthermore, similar equalities hold for $\mathrm{Err}(C^* \boxdot c)$ *and* $\mathrm{Err}(C \boxdot c^*)$.

5.6 Key switching

Key switching is a technique that changes a RLWE-ciphertext (or RLWE*-ciphertext) of the same message encrypted under one key to another RLWE-ciphertext (or RLWE*-ciphertext) encrypted under another key, without decrypting the message. This process uses a *key switching key*: the key switching key, written as $\mathrm{KSK}_{s_1 \to s_2}$, is created by encrypting the first key s_1 using the second key s_2. This is a standard method described in many research papers (see, for example, [15, 16, 21]), and for our purposes, we will just assume we have a function called

$$\mathrm{KeySwitch}_{s_1(X) \to s_2(X)}$$

that performs this operation[2].

6 Extraction of a LWE from a RLWE

This section describes a procedure aimed at extracting an LWE-encryption $\mathrm{LWE}_{\mathbf{s}}(\mu_i)$ of the i^{th} coefficient $\mu_i \in \mathbb{T}$ from a RLWE-ciphertext $\mathrm{RLWE}_s(\mu)$, where the polynomial message is expressed as

$$\mu(X) = \sum_{i=0}^{N-1} \mu_i \Omega_i(X),$$

where $(\Omega_i(X))_{0 \le i < N} = (X^i)_{0 \le i < N}$ but which could be any basis of $\mathcal{R}$. The connection between s and $\mathbf{s}$ will be elucidated in the subsequent discussion.

6.1 An extraction procedure using dual bases

The extraction operation can be articulated using the scalar product: from the polynomial message $\mu(X)$, the coefficient μ_i can be obtained as

$$\mu_i = \langle \Omega_i^*(X), \mu(X) \rangle$$

This expression is equal to μ_i by definition of Ω_i^*. Now, considering that

$$\mu(X) = b(X) - s(X) \cdot a(X) - e(X) \bmod \mathcal{R},$$

that is to say that $\mu(X)$ is given in RLWE-encrypted form $c(X) = (a(X), b(X)) \in \mathcal{T} \times \mathcal{T}$, we have the following relations:

$$\begin{aligned}
\mu_i = \langle \Omega_i^*(X), \mu(X) \rangle &= \langle \Omega_i^*(X), b(X) \rangle - \langle \Omega_i^*(X), s(X) \cdot a(X) \rangle - \langle \Omega_i^*(X), e(X) \rangle \\
&= \langle \Omega_i^*(X), b(X) \rangle - \sum_{j=0}^{N-1} s_j \langle \Omega_i^*(X), B_j(X) \cdot a(X) \rangle - \langle \Omega_i^*(X), e(X) \rangle \\
&:= b^{(i)} - \mathbf{s} \cdot \mathbf{a}^{(i)} - e^{(i)}
\end{aligned}$$

where we have assumed that $s(X)$ is written in a basis $(B_j)_{0 \le j < N}$ as

$$s(X) = \sum_{j=0}^{N-1} s_j B_j(X) \quad \text{and} \quad \mathbf{s}_j = s_{j-1} \quad \text{for} \quad j = 1, \ldots, N. \tag{6}$$

[2]We don't need to go into the details of how it works internally, as the process is the same regardless of the specific cyclotomic polynomial used in the encryption scheme.

The values $b^{(i)}$, $\mathbf{a}^{(i)}$ and $e^{(i)}$ are thus defined as

$$b^{(i)} = \langle \Omega_i^*(X), b(X) \rangle, \quad \mathbf{a}_j^{(i)} = \langle \Omega_i^*(X), B_{j-1}(X) \cdot a(X) \rangle \quad \text{for} \quad j = 1, \ldots, N,$$

and $e^{(i)} = \langle \Omega_i^*(X), e(X) \rangle$. Thus, we obtain an LWE-encryption of μ_i as

$$\mathrm{LWE}_{\mathbf{s}}(\mu_i) = (\mathbf{a}^{(i)}, b^{(i)}).$$

In order to get explicit expressions, we can assume for instance, that $B_i(X) = \Omega_i(X) = X^i$ and that $a(X) = \sum_{j=0}^{N-1} a_j \Omega_j(X)$, $b(X) = \sum_{j=0}^{N-1} b_j \Omega_j(X)$. Using the expression of $s(X) \cdot a(X) \bmod \Phi_M$ furnished in Lemma 9.2, we then obtain

$$\mathbf{a}_j^{(i)} = a_{i-j+1} - a_{N-j+1+[i]_{\frac{M}{t}}} \quad \text{for} \quad j = 1, \ldots, N,$$

and a LWE-encryption of μ_i as

$$\mathrm{LWE}_{\mathbf{s}}(\mu_i) = (\mathbf{a}^{(i)}, b_i).$$

The dual version of this procedure can be derived in a similar fashion: given an RLWE*-encryption $(a^*(X), b^*(X))$ of the polynomial message

$$\mu^*(X) = \sum_{j=0}^{N-1} \mu_j^* \Omega_j^*(X),$$

the coefficient μ_i^* can be obtained as

$$\begin{aligned} \mu_i^* &= \langle \Omega_i(X), \mu^*(X) \rangle \\ &= \langle \Omega_i(X), b^*(X) \rangle - \sum_{j=0}^{N-1} s_j \langle \Omega_i(X), B_j(X) \cdot a^*(X) \rangle - \langle \Omega_i(X), e^*(X) \rangle \\ &:= b^{(i)} - \mathbf{s} \cdot \mathbf{a}^{(i)} - e^{(i)} \end{aligned}$$

where we have assumed relations (6). The values $b^{(i)}$, $\mathbf{a}^{(i)}$ and $e^{(i)}$ are thus defined as

$$b^{(i)} = \langle \Omega_i(X), b^*(X) \rangle, \quad \mathbf{a}_j^{(i)} = \langle \Omega_i(X), B_{j-1}(X) \cdot a^*(X) \rangle \quad \text{for} \quad j = 1, \ldots, N,$$

and $e^{(i)} = \langle \Omega_i(X), e^*(X) \rangle$. Thus, we obtain an LWE-encryption of μ_i as

$$\mathrm{LWE}_{\mathbf{s}}(\mu_i) = (\mathbf{a}^{(i)}, b^{(i)}).$$

Again, explicit expressions can be obtained by assuming for instance, that $B_i(X) = \Omega_i(X) = X^i$ and that $a^*(X) = \sum_{j=0}^{N-1} a_j^* \Omega_j^*(X)$, $b^*(X) = \sum_{j=0}^{N-1} b_j^* \Omega_j^*(X)$. In that case

$$\mathbf{a}_j^{(i)} = \langle X^{i-j+1}, a^*(X) \rangle$$

and using the expression of $X^{i-j+1} \bmod \Phi_M$ furnished in Lemma 9.2, we finally obtain

$$\begin{array}{ll} \text{for } 1 \leq j \leq i+1: & \mathbf{a}_j^{(i)} = a_{i-j+1}^* \\ \text{for } i+2 \leq j \leq \frac{M}{t} + i + 1: & \mathbf{a}_j^{(i)} = -\sum_{\ell=1}^{t-1} a_{\ell\frac{M}{t}+i-j+1}^* \\ \text{for } \frac{M}{t} + i + 2 \leq j \leq N: & \mathbf{a}_j^{(i)} = a_{M+i-j+1}^* \end{array}$$

and a LWE-encryption of μ_i^* as

$$\mathrm{LWE}_{\mathbf{s}}(\mu_i^*) = (\mathbf{a}^{(i)}, b_i^*).$$

Proposition 6.1 *Given an encryption* $(a(X), b(X)) \in \mathcal{T} \times \mathcal{T}$ *of*

$$\mu(X) = \sum_{j=0}^{N-1} \mu_j \Omega_j(X)$$

with key $s(X) = \sum_{j=0}^{N-1} s_j B_j(X)$ *and an integer index* $i \in \{0, \ldots, N-1\}$*, a* LWE*-encryption with key* $\mathbf{s} = (s_0, \ldots, s_{N-1})$ *of* μ_i *is obtained as*

$$\mathrm{LWE}_{\mathbf{s}}(\mu_i) = (\mathbf{a}^{(i)}, b^{(i)}),$$

where

$$b^{(i)} = \langle \Omega_i^*(X), b(X) \rangle, \quad \mathbf{a}_j^{(i)} = \langle \Omega_i^*(X), B_{j-1}(X) \cdot a(X) \rangle \quad \textit{for} \quad j = 1, \ldots, N.$$

If we furthermore assume that $\Omega_i(X) = B_i(X) = X^i$ *for* $0 \leq i < N$ *and that*

$$a(X) = \sum_{j=0}^{N-1} a_j X^j \quad \textit{and} \quad b(X) = \sum_{j=0}^{N-1} b_j X^j,$$

then

$$b^{(i)} = b_i \quad \textit{and} \quad \mathbf{a}_j^{(i)} = a_{i-j+1} - a_{N-j+1+[i]_{\frac{M}{t}}} \quad \textit{for} \quad j = 1, \ldots, N.$$

Similarly, given an encryption $(a^*(X), b^*(X)) \in \mathcal{T}^\vee \times \mathcal{T}^\vee$ *of*

$$\mu^*(X) = \sum_{i=0}^{N-1} \mu_i^* \Omega_i^*(X)$$

with key $s(X) = \sum_{j=0}^{N-1} s_j B_j(X)$ *and an integer index* $i \in \{0, \ldots, N-1\}$*, a* LWE*-encryption with key* $\mathbf{s} = (s_0, \ldots, s_{N-1})$ *of* μ_i^* *is obtained as*

$$LWE_{\mathbf{s}}(\mu_i^*) = (\mathbf{a}^{(i)}, b^{(i)}),$$

where

$$b^{(i)} = \langle \Omega_i(X), b^*(X) \rangle \quad \textit{and} \quad \mathbf{a}_j^{(i)} = \langle \Omega_i(X), B_{j-1}(X) \cdot a^*(X) \rangle \quad \textit{for} \quad j = 1, \ldots, N.$$

If we furthermore assume that $\Omega_i(X) = B_i(X) = X^i$ *for* $0 \leq i < N$ *and that*

$$a^*(X) = \sum_{j=0}^{N-1} a_j^* \Omega_j^*(X) \quad \textit{and} \quad b^*(X) = \sum_{j=0}^{N-1} b_j^* \Omega_j^*(X),$$

then

$$b^{(i)} = b_i^* \quad \textit{and} \quad \begin{cases} \mathbf{a}_j^{(i)} = a_{i-j+1}^* & \textit{for } 1 \leq j \leq i+1 \\ \mathbf{a}_j^{(i)} = -\sum_{\ell=1}^{t-1} a_{\ell\frac{M}{t}+i-j+1}^* & \textit{for } i+2 \leq j \leq \frac{M}{t}+i+1 \\ \mathbf{a}_j^{(i)} = a_{M+i-j+1}^* & \textit{for } \frac{M}{t}+i+2 \leq j \leq N. \end{cases} \tag{7}$$

Remark 6.2 *Alternatively, we could have expressed*

$$\begin{aligned}\mu_i = \langle \Omega_i^*(X), \mu \rangle &= \langle \Omega_i^*(X), b(X) \rangle - \langle \Omega_i^*(X), s(X) \cdot a(X) \rangle - \langle \Omega_i^*(X), e(X) \rangle \\ &= b_i - \mathbf{s} \cdot \mathbf{a} - e_i\end{aligned}$$

by assuming

$$a(X) = \sum_{j=0}^{N-1} a_j \Omega_j(X) \quad \textit{and} \quad b(X) = \sum_{j=0}^{N-1} b_j \Omega_j(X),$$

and writing

$$\langle \Omega_i^*(X), s(X) \cdot a(X) \rangle = \sum_{j=0}^{N-1} a_j \langle \Omega_i^*(X), \Omega_j(X) s(X) \rangle,$$

so that

$$\mathbf{a}_j = a_{j-1} \quad \textit{and} \quad \mathbf{s}_j = \langle \Omega_i^*(X), \Omega_{j-1}(X) s(X) \rangle \quad \textit{for} \quad j = 1, \ldots, N.$$

This corresponding encryption is characterized by the use of a secret key with components drawn from a larger set than the original set $\mathbb{S}$, which could be a disadvantage in certain implementations.

Algorithm 11 LWE extraction of μ_i from $\mu(X) = \sum_{j=0}^{N-1} \mu_j \Omega_j(X) \in \mathcal{T}$

Input: $(a(X), b(X)) = \mathrm{RLWE}_s(\mu(X)) \in \mathcal{T} \times \mathcal{T}$, index $i \in \{0, \ldots, N-1\}$.

1: **for** each j from 1 to N **do**
2: $\quad \mathbf{a}_j \leftarrow a_{i-j+1} - a_{N-j+1+[i]_{\frac{M}{t}}}$
3: $\quad \mathbf{s}_j \leftarrow s_{j-1}$
4: **end for**
5: $b_i \leftarrow (b(X))_i$
6: **Return:** $\mathrm{LWE}_{\mathbf{s}}(\mu_i) = (\mathbf{a}, b_i)$

Note: $a(X)$, $b(X)$, $s(X)$ are expressed in $(\Omega_j)_{0 \le j < N}$

Algorithm 12 LWE extraction of μ_i^* from $\mu^*(X) = \sum_{j=0}^{N-1} \mu_j^* \Omega_j^*(X) \in \mathcal{T}^\vee$

Input: $(a^*(X), b^*(X)) = \mathrm{RLWE}_s^*(\mu^*(X)) \in \mathcal{T}^\vee \times \mathcal{T}^\vee$, index $i \in \{0, \ldots, N-1\}$.

1: **for** each j from 1 to N **do**
2: $\quad \mathbf{a}_j \leftarrow \langle X^{i+M-j+1}, a^*(X) \rangle$
3: $\quad \mathbf{s}_j \leftarrow s_{j-1}$
4: **end for**
5: $b_i^* \leftarrow (b^*(X))_i$
6: **Return:** $\mathrm{LWE}_{\mathbf{s}}(\mu_i) = (\mathbf{a}, b_i^*)$

Note: $a^*(X)$, $b^*(X)$ expressed in $(\Omega_j^*)_{0 \le j < N}$ and $s(X)$ in $(\Omega_j)_{0 \le j < N}$

6.2 Blind extraction using registers

If the index i of the coefficient we want to extract from

$$(i): \ \mu(X) \in \mathcal{T} \quad \text{or} \quad (ii): \ \mu^*(X) \in \mathcal{T}^\vee$$

is not provided as a specific value but instead as an encrypted index, then the previous procedure becomes ineffective. In this subsection, we consider the scenario where i is represented as an RGSW*-encryption of Ω_i^* (in case (i)) or as an RGSW-encryption of Ω_i (in case (ii)). These RGSW-encryptions are referred to as registers in [33].

In the first scenario, we write

$$\mu_i = \langle \Omega_i^*(X), \mu \rangle = \mathrm{Tr}\left(\bar{\Omega}_i^*(X)\mu(X)\right)$$

where now both $\mu(X) \in \mathcal{T}$ and $\bar{\Omega}_i^*(X) \in \mathcal{R}^\vee$ are encrypted as

$$c(X) = (a(X), b(X)) = \mathrm{RLWE}_s(\mu(X)) \in \mathcal{T} \times \mathcal{T}$$

and

$$C^*(X) = (a_j^*(X), b_j^*(X))_{1 \le j \le 2\ell} = \mathrm{RGSW}_s^*(\bar{\Omega}_i^*(X)) \in (\mathcal{T}^\vee \times \mathcal{T}^\vee)^{2\ell}.$$

A homomorphic external product results in:

$$c^*(X) = (a^*(X), b^*(X)) = C^*(X) \boxdot c(X) = \mathrm{RLWE}_s^*(\bar{\Omega}_i^*(X)\mu(X))$$

It then remains to obtain a LWE-encryption of the trace of c^* and this can be done as in Proposition 6.1.

In the dual version, we express

$$\mu_i^* = \langle X^i, \mu^* \rangle = \mathrm{Tr}\left(X^{-i}\mu^*(X)\right)$$

where now both $\mu^*(X) \in \mathcal{T}^\vee$ and $X^{-i} \in \mathcal{R}$ are encrypted as

$$c^*(X) = (a^*(X), b^*(X)) = \mathrm{RLWE}_s^*(\mu^*(X)) \in \mathcal{T}^\vee \times \mathcal{T}^\vee$$

and

$$C(X) = (a_j(X), b_j(X))_{1 \le j \le 2\ell} = \mathrm{RGSW}_s(X^{-i}) \in (\mathcal{T} \times \mathcal{T})^{2\ell}.$$

An homomorphic external product then gives

$$c^*(X) = (a^*(X), b^*(X)) = C(X) \boxdot c^*(X) = \mathrm{RGSW}_s(X^{-i}\mu(X))$$

and it remains to get a LWE-encryption of the trace of c^*: to this aim, we proceed as in Proposition 6.1.

6.2.1 Simple extraction

Assume $s(X) = \sum_{j=0}^{N-1} s_j \Omega_j(X)$ and let $(B_i(X))_{0 \le i \le N-1}$ be a basis of $\mathcal{R}$ and $(B_i^*(X))_{0 \le i \le N-1}$ its dual w.r.t. $\langle \cdot, \cdot \rangle$.

Objective: from an encryption $C^*(X) = \mathrm{RGSW}^*_{s(X)}(\bar{B}_i^*(X))$ of an index $0 \le i < N$ and an encryption $c(X) = \mathrm{RLWE}_{s(X)}(\mu(X))$ of the polynomial $\mu(X) = \sum_{i=0}^{N-1} \mu_i B_i(X)$, get a LWE-encryption of μ_i.

Procedure: (i) compute $c^*(X) = (a^*(X), b^*(X)) = C^*(X) \boxdot c(X) = \mathrm{RLWE}^*_{s(X)}(\bar{B}_i^*(X)\mu(X))$; (ii) write the corresponding LWE-encryption. To this aim, we just write $\mu_i = \langle B_i^*(X), \mu(X) \rangle\rangle$ in encrypted form (recall that $\bar{B}_i^*(X)\mu(X) = b^*(X) - s(X)a^*(X) - e^*(X)$):

$$\mu_i = \langle B_i^*(X), \mu(X) \rangle\rangle = \mathrm{Tr}(b^*(X)) - \sum_{j=0}^{N-1} s_j \langle \bar{\Omega}_j(X), a^*(X) \rangle\rangle - \mathrm{Tr}(e^*(X))$$

so that

$$\mathrm{LWE}_{\mathbf{s}}(\mu_i) = (\mathbf{a}, b) \quad \text{with} \quad \mathbf{a}_j = \langle \bar{\Omega}_{j-1}(X), a^*(X) \rangle,\ j = 1, \ldots, N \text{ and } b = \mathrm{Tr}(b^*(X)).$$

If $a^*(X)$ and $b^*(X)$ are, for instance, both expressed in the basis Ω^* as $a^*(X) = \sum_{k=0}^{N-1} a_k^* \Omega_k^*(X)$ and $b^*(X) = \sum_{k=0}^{N-1} a_k^* \Omega_k^*(X)$, then we have $b = b_0^*$ and

$$\begin{aligned} \mathbf{a}_j = \sum_{k=0}^{N-1} a_k^* \langle X^{M-j+1}, \Omega_k^*(X) \rangle = a_{M-j+1}^* & \qquad \text{for } \frac{M}{t} + 1 < j \leq N, \\ = -\sum_{\ell=1}^{t-1} a_{\ell\frac{M}{t}-j+1}^* & \qquad \text{for } 1 \leq j \leq \frac{M}{t} + 1. \end{aligned}$$

Algorithm 13 Blind Extraction of Coefficient μ_i of $\mu(X) \in \mathcal{T}$

Input: Encryption $c(X) \in \mathcal{T} \times \mathcal{T}$ of $\mu(X) = \sum_{j=0}^{N-1} \mu_j B_j(X)$ and encryption $C^*(X) = \mathrm{RGSW}^*_{s(X)}(\bar{B}_i^*(X))$ of an index $0 \leq i < N$.

1: $c^*(X) = (a^*(X), b^*(X)) \leftarrow C^*(X) \boxdot c(X)$
2: $b \leftarrow \mathrm{Tr}(b^*(X))$
3: **for** $j = 1$ **to** N **do**
4: $\quad \mathbf{a}_j \leftarrow \langle \bar{\Omega}_{j-1}(X), a^*(X) \rangle$,
5: $\quad \mathbf{s}_j \leftarrow s_{j-1}$
6: **end for**
7: **Return** $\mathrm{LWE}_{\mathbf{s}}(\mu_i) = (\mathbf{a}, b)$.

6.2.2 Representation of a linear form of a vector in $\mathbb{T}^N$.

Assume $s(X) = \sum_{j=0}^{N-1} s_j \Omega_j(X)$ and let $(B_i(X))_{0 \leq i \leq N-1}$ be a basis of $\mathcal{R}$ and $(B_i^*(X))_{0 \leq i \leq N-1}$ its dual w.r.t. $\langle \cdot, \cdot \rangle$. Let $\ell \in \mathcal{M}_{1,N-1}(\mathbb{Z})$ be a linear form from $\mathbb{T}^{N-1}$ to $\mathbb{T}$:

$$\mu \in \mathbb{T}^N \mapsto \ell^T \mu = \sum_{i=0}^{N-1} \ell_i \mu_i \in \mathbb{T}.$$

Objective: from an encryption $C^*(X) = \mathrm{RGSW}^*_{s(X)}(L^*(X))$ of a well-chosen polynomial $L^*(X)$ and an encryption $c(X) = \mathrm{RLWE}_{s(X)}(\mu(X))$ of the polynomial $\mu(X) = \sum_{i=0}^{N-1} \mu_i B_i(X)$, get a LWE-encryption of $\ell^T \mu$.

Procedure: (i) compute $c^*(X) = (a^*(X), b^*(X)) = C^*(X) \boxdot c(X) = \mathrm{RLWE}^*_{s(X)}(\bar{L}^*(X)\mu(X))$; (ii) write the corresponding LWE-encryption.

If C^* is built to encrypt

$$L^*(X) = \sum_{i=0}^{N-1} \ell_i B_i^*(X)$$

then we get

$$\langle L^*(X), \mu(X) \rangle \rangle = \sum_{i=0}^{N-1} \ell_i \langle B_i^*(X), \mu(X) \rangle \rangle = \sum_{i=0}^{N-1} \ell_i \mu_i.$$

The corresponding Algorithm is given below:

Algorithm 14 Blind Extraction of $\sum_{i=0}^{N-1} \ell_i \mu_i$ from $\mu(X) \in \mathcal{T}$

Input: Encryption $c(X) \in \mathcal{T} \times \mathcal{T}$ of $\mu(X) = \sum_{j=0}^{N-1} \mu_j B_j(X)$ and encryption $C^*(X) = \mathrm{RGSW}^*_{s(X)}(\bar{L}^*(X))$ with $L^*(X) = \sum_{i=0}^{N-1} \ell_i B_i^*(X)$.

1: $c^*(X) = (a^*(X), b^*(X)) \leftarrow C^*(X) \boxdot c(X)$
2: $b \leftarrow \mathrm{Tr}(b^*(X))$
3: **for** $j = 1$ **to** N **do**
4: $\quad \mathbf{a}_j \leftarrow \langle \bar{\Omega}_{j-1}(X), a^*(X) \rangle$,
5: $\quad \mathbf{s}_j \leftarrow s_{j-1}$
6: **end for**
7: **Return** $\mathrm{LWE}_{\mathbf{s}}(\ell^T \mu) = (\mathbf{a}, b)$.

Note that a linear map from $\mathbb{T}^N$ to $\mathbb{T}^V$ can be viewed as a set of V linear forms, and it can be "encrypted" similarly. This corresponds to the linear component of a neural network layer where both the input data and the weights are encrypted (possibly by two different persons).

7 Bootstrapping in the prime power cyclotomic setting

The primary objective of bootstrapping in fully homomorphic encryption (FHE) is to reduce the noise that builds up in ciphertext during operations, as excessive noise can impede decryption. The key objectives of bootstrapping can be summarized as follows:

(i) **Noise Reduction**: Enables an unlimited number of homomorphic operations by refreshing the ciphertext and lowering the noise level;

(ii) **Function Mapping**: Facilitates the application of a function during the bootstrapping process;

(iii) **Security Maintenance**: Ensures that the original data remains secure throughout the operation.

In the context of TFHE, this goal can be articulated as follows:

Goal of bootstrapping: Given a LWE-encryption c of $\mu \in \mathbb{T}_p$ with error e and a function f that maps $\mathbb{T}_p$ to $\mathbb{T}$, produce a LWE-encryption of $f(\mu)$ with a fresh error of smaller size than e.

In summary, bootstrapping is crucial for the practical implementation of encrypted computations in HE. A significant contribution of the authors of TFHE was the improvement of the efficiency of their variant of FHEW bootstrapping, highlighting the need to preserve this efficiency in the context of prime power cyclotomic polynomials. Therefore, it becomes essential to reformulate the mathematical representation of bootstrapping in a manner that aligns with our context. The following function serves as the foundation for our formulation of the bootstrapping process:

Definition 7.1 *Given a polynomial $v^*(X) \in \mathcal{T}^\vee$, the bootstrap function Θ_{v^*} is defined as*

$$\begin{array}{rccl} \Theta_{v^*}: & \mathbb{T} & \to & \mathbb{T} \\ & \mu & \mapsto & \mathrm{Tr}\Big(X^{-\lfloor M\mu \rceil} v^*(X)\Big) = \langle X^{\lfloor M\mu \rceil}, v^*(X) \rangle \end{array}$$

It is important to highlight that this definition and its implementation retain the concepts of the bootstrapping procedure proposed by Ducas and Micciancio [21], with the distinction lying solely in its formulation. It is evident that Θ_{v^*} is fully characterized by the values

$$\Theta_{v^*}\Big(\frac{i}{M}\Big), \quad 0 \leq i \leq M-1,$$

along with the observation that

$$\forall \mu \in \mathbb{T}, \quad \Theta_{v^*}(\mu) = \Theta_{v^*}\Big(\lfloor M\mu \rceil\Big).$$

A homomorphic and efficient implementation of this function, essential for the correctness of the bootstrapping procedure, must satisfy the following:

Requirement: For all $\mu \in \mathbb{T}_p$ and for all error $e \in \mathbb{T}$ such that $|e| \leq \frac{1}{2p}$

$$\Theta_{v^*}\Big(\mu + e\Big) = f(\mu). \tag{8}$$

Note that condition (8) implies that

$$\forall \mu \in \mathbb{T}, \quad \Theta_{v^*}\Big(\mu\Big) = f\Big(\frac{\lfloor p\mu \rceil}{p}\Big).$$

In this section, we aim to establish clear and concise compatibility conditions on f that guarantee the existence of v^* and explicitly determine its form.

7.1 General formulation of the compatibility conditions

Let the function F be defined on the interval $0 \leq i \leq M-1$ by

$$F(i) = \Theta_{v^*}\Big(\frac{i}{M}\Big) = \mathrm{Tr}(X^{-i}v^*(X)), \quad 0 \leq i \leq M-1,$$

that is to say

$$F(i) = \Theta_{v^*}\Big(\frac{i}{M}\Big) = \langle X^i, v^*(X)\rangle, \quad 0 \leq i \leq M-1.$$

The coefficients of the polynomial

$$v^*(X) = \sum_{i=0}^{N-1} v_i^* \Omega_i^*(X) \in \mathcal{T}^\vee$$

are completely and uniquely determined by the relations

$$F(i) = \Theta_{v^*}\Big(\frac{i}{M}\Big) = \langle X^i, v^*(X)\rangle, \quad 0 \leq i \leq N-1,$$

which leads to

$$v_i^* = F(i), \quad 0 \leq i \leq N-1.$$

However, the values of $F(i)$ for $N \leq i \leq M-1$ are then constrained by

$$F(i) = \langle X^i, v^*(X)\rangle, \quad N \leq i \leq M-1.$$

For $N \leq i < M$, we can express X^i as

$$X^i = \sum_{j=0}^{N-1} \beta_{ij} X^j.$$

Under this condition, we find that

$$\forall N \leq i \leq M-1, \quad F(i) = \sum_{j,k=0}^{N-1} \beta_{ij} v_k^* \langle X^j, \Omega_k^* \rangle = \sum_{j=0}^{N-1} \beta_{ij} F(j)$$

This results in $M - N$ compatibility conditions that may be equivalently expressed in terms of f as

$$\forall N \leq i \leq M-1, \quad f\Big(\frac{\lfloor p\frac{i}{M} \rceil}{p}\Big) = \sum_{j=0}^{N-1} \beta_{ij} \quad f\Big(\frac{\lfloor p\frac{j}{M} \rceil}{p}\Big).$$

7.2 Explicit computation of the test polynomial

It is easy to check from Lemma 9.1 that

$$\forall N \leq i \leq M-1, \quad X^i = - \sum_{\substack{0 \leq j \leq N-1, \\ [j]_{M/t} = i - N}} X^j,$$

so that

$$\beta_{ij} = -\delta_{i,N+[j]_{M/t}}$$

where, as is standard, $\delta_{i,j} = 1$ if $i = J$ and 0 otherwise. The compatibility relations thus write

$$N \leq i \leq M-1, \quad F(i) + \sum_{\substack{0 \leq j \leq N-1, \\ [j]_{M/t} = i - N}} F(j) = 0$$

or equivalently

$$\forall 0 \leq r \leq \frac{M}{t} - 1, \quad \sum_{k=0}^{t-1} F\Big(kt^{\alpha-1} + r\Big) = 0.$$

In summary, we can state the following:

Proposition 7.2 *There exists a polynomial $v^* \in \mathcal{T}^\vee$ such that*

$$\forall \mu \in \mathbb{T}_p, \quad \Theta_{v^*}\Big(\mu + e\Big) = f(\mu)$$

for all errors $e \in \mathbb{T}$ with $|e| \leq \frac{1}{2p}$ if and only if the function f satisfies the compatibility conditions

$$\forall 0 \leq r \leq \frac{M}{t} - 1, \quad \sum_{k=0}^{t-1} f\left(\frac{\lfloor (kt^{\alpha-1} + r)\frac{p}{M} \rceil}{p}\right) = 0. \tag{9}$$

The polynomial v^* are then determined by the relation

$$v^*(X) = \sum_{i=0}^{N-1} f\Big(\frac{\lfloor p\frac{i}{M} \rceil}{p}\Big) \Omega_i^*(X).$$

Remark 7.3 *In the specific case where $p = t$, the compatibility relations simplify to a single equation:*

$$\sum_{k=0}^{t-1} f\left(\frac{k}{t}\right) = 0.$$

This can be understood by noting that for any integer $\ell \in \mathbb{Z}$, it holds that

$$\sum_{k=0}^{t-1} f\left(\frac{k}{t} + \frac{\ell}{t}\right) = \sum_{k=0}^{t-1} f\left(\frac{k}{t}\right).$$

It is important to note that this condition can be easily eliminated by adding a constant to f prior to bootstrapping and subsequently removing it from the result.

7.3 Homomorphic implementation

We now convert Definition 7.1 into a practical algorithm that operates on the ciphertext $(\mathbf{a}, b)$, which encrypts $\mu \in \mathbb{T}_p$ with a secret key $\mathbf{s}$:

$$\mu = b - \mathbf{s} \cdot \mathbf{a} - e.$$

The first step, while not critical to the definition of bootstrapping, is essential for enhancing its efficiency. This step involves re-encrypting the ciphertext with a smaller key $\hat{\mathbf{s}} \in \mathbb{S}^{\hat{n}}$, such that

$$\mu = \hat{b} - \hat{\mathbf{s}} \cdot \hat{\mathbf{a}} - \hat{e}.$$

Bootstrapping fundamentally entails the homomorphic computation of an LWE-encryption of $\Theta_{v^*}(\mu)$ from an LWE-encryption of μ. Since rounding to an integer is not inherently a homomorphic operation, it is necessary to first approximate

$$\lfloor M\mu \rceil = \lfloor M(\hat{b} - \hat{\mathbf{s}} \cdot \hat{\mathbf{a}} - \hat{e}) \rceil$$

To achieve this, we employ the *collapsing* strategy from [9]. For the sake of simplicity, we suppose here that m divides $\hat{n}$. We denote, on the one hand,

$$\tilde{\mathbf{s}}_k = \left(\hat{s}_{m(k-1)+1}, \hat{s}_{m(k-1)+2}, \ldots, \hat{s}_{mk}\right) \in \mathbb{S}^m, \quad k = 1, \ldots, \hat{n}/m, \tag{10}$$

and on the other hand

$$\tilde{\mathbf{a}}_k = \left(\hat{a}_{m(k-1)+1}, \hat{a}_{m(k-1)+2}, \ldots, \hat{a}_{mk}\right) \in \mathbb{T}_q^m, \quad k = 1, \ldots, \hat{n}/m, \tag{11}$$

so that

$$\mu + e = \hat{b} - \hat{\mathbf{s}} \cdot \hat{\mathbf{a}} = \hat{b} - \sum_{k=1}^{\hat{n}/m} \tilde{\mathbf{s}}_k \cdot \tilde{\mathbf{a}}_k.$$

As explained in [9], we then round partial sums $\tilde{\mathbf{s}}_k \cdot \tilde{\mathbf{a}}_k$ and not individual products $s_k a_k$ as it is customary [15, 16, 21]. This leads to the approximation

$$\lfloor M(\mu + e) \rceil \approx -\sum_{k=1}^{\hat{n}/m} \sum_{\tilde{\mathbf{j}} \in \mathbb{S}^m} \delta_{\tilde{\mathbf{j}}, \tilde{\mathbf{s}}_k} \bar{a}_{k, \tilde{\mathbf{j}}} = -\sum_{k=1}^{\hat{n}/m} \bar{a}_{k, \tilde{\mathbf{s}}_k} =: \imath, \tag{12}$$

where we denote $\delta_{\tilde{\mathbf{i}},\tilde{\mathbf{j}}}$, for $(\tilde{\mathbf{i}},\tilde{\mathbf{j}}) \in \mathbb{S}^m \times \mathbb{S}^m$, the symbol with value 1 if $\tilde{\mathbf{i}} = \tilde{\mathbf{j}}$ and 0 otherwise, and where

$$\bar{a}_{1,\tilde{\jmath}} = \lfloor M\tilde{\mathbf{a}}_1 \cdot \tilde{\mathbf{j}} - Mb \rceil, \qquad \bar{a}_{k,\tilde{\mathbf{j}}} = \lfloor M\tilde{\mathbf{a}}_k \cdot \tilde{\mathbf{j}} \rceil \text{ for } k = 2, \ldots, \hat{n}/m \text{ and } \tilde{\mathbf{j}} \in \mathbb{S}^m. \tag{13}$$

Note that the sum in (12) is valid for all m dividing $\hat{n}$, in particular for $m = 1$, where we recover the usual expression, as seen in [15], for example, or for $m = \hat{n}$, where the two sides (of the approx sign) in equation (12) become equal. We finally observe that

$$X^{-\iota} = \prod_{k=1}^{\hat{n}/m} X^{\bar{a}_{k,\tilde{\mathbf{s}}_k}} = \prod_{k=1}^{\hat{n}/m} \sum_{\tilde{\mathbf{j}} \in \mathbb{S}^m} \delta_{\tilde{\mathbf{j}},\tilde{\mathbf{s}}_k} X^{\bar{a}_{k,\tilde{\jmath}}} = \prod_{k=1}^{\hat{n}/m} H_k(X) \tag{14}$$

with

$$H_k(X) = \sum_{\tilde{\mathbf{j}} \in \mathbb{S}^m} \delta_{\tilde{\mathbf{j}},\tilde{\mathbf{s}}_k} X^{\bar{a}_{k,\tilde{\jmath}}} \in \mathcal{R}, \quad k = 1, \ldots, \hat{n}/m, \tag{15}$$

so that $X^{-\iota} \cdot v^*(X)$ can be computed as the result of $\hat{n}/m$ successive modular products $\mathcal{R} \times \mathcal{T}^\vee$ applied from the right to the left

$$X^{-\iota} \cdot v^*(X) = H_{n/m}(X) \ldots (H_2(X) \cdot (H_1(X) \cdot v^*(X))) \ldots) \tag{16}$$

The full bootstrapping procedure involves three steps:

1. **Keyswitch Operation:** Given $\mathbf{c} = (\mathbf{a}, b)$ a $\mathrm{LWE}_{\mathbf{s}}$-encryption of a message $\mu \in \mathbb{T}_p$, compute $\hat{\mathbf{c}} = (\hat{\mathbf{a}}, \hat{b})$ a $\mathrm{LWE}_{\hat{\mathbf{s}}}$-encryption for a reduced size key $\hat{\mathbf{s}}$;
2. **Blind Rotate Operation:** Given RGSW_s-encryptions of the $\delta_{\tilde{\mathbf{j}},\tilde{\mathbf{s}}_k}$, computes a RLWE^*_s-encryption of $X^{-\iota} \cdot v^*(X)$;
3. **Extract Operation:** Compute an $\mathrm{LWE}_{\mathbf{s}}$-encryption of the trace of $X^{-\iota} \cdot v^*(X)$, which is the final output of the bootstrap[3].

The first step is entirely standard and resembles for instance what is done in the context of powers-of-two cyclotomic polynomials; therefore, we will omit its description. Below, we will outline the two other essential steps in detail.

Blind rotation

Given RGSW_s-encryptions of the $\delta_{\tilde{\mathbf{j}},\tilde{\mathbf{s}}_k}$, it is straightforward to compute the homomorphic RGSW_s-encryptions of $H_k(X)$ for $k = 1, \ldots, \hat{n}/m$,as outlined in Formula (15). Specifically, we have:

$$\mathrm{RGSW}_s(H_k) = \bigoplus_{\tilde{\mathbf{j}} \in \mathbb{S}^m} \left(X^{\bar{a}_{k,\tilde{\mathbf{j}}}} \cdot \mathrm{RGSW}_s(\delta_{\tilde{\mathbf{j}},\tilde{\mathbf{s}}_k}) \right), \tag{17}$$

where the $\cdot$ denotes the product of the polynomial $X^{\bar{a}_{k,\tilde{\mathbf{j}}}} \in \mathcal{R}$ by each of the polynomial components of $\mathrm{RGSW}_s(\delta_{\tilde{\mathbf{j}},\tilde{\mathbf{s}}_k})$, all of which reside in $\mathcal{T}^\vee$. The RLWE^*_s-encrypted value $X^{-\iota} \cdot v^*(X)$ can now be computed homomorphically according to Formula (16) as follows:

$$\mathrm{RGSW}_s(H_{n/m}) \boxdot (\ldots (\mathrm{RGSW}_s(H_1) \boxdot \mathrm{RLWE}^*_s(v^*)) \ldots). \tag{18}$$

Note that $\mathrm{RLWE}^*_s(v^*)$ is defined here as the trivial noise-free zero-mask $(0, \ldots, 0, v^*(X))$. Additionally, we assume that v^* aligns with its definition in Proposition 7.2 for the specified target function f.

[3]The final LWE-key is fully specified by the polynomial key $s(X) = \sum_{j=0}^{N-1} s_j X^j \in \mathcal{R}$, and the extraction procedure we adopt guarantees that $\mathbf{s} = (s_0, s_1, \ldots, s_{N-1})$.

Algorithm 15 Blind Rotation Algorithm

Input: $\mathbf{c} = (\mathbf{a}, b) \in \mathbb{T}^{\hat{n}+1}$ and $\mathrm{RGSW}_s(\delta_{\tilde{\mathbf{j}},\tilde{\mathbf{s}}_k})$ for $\tilde{\mathbf{j}} \in \mathbb{S}^m$, $k = 1, \ldots, \hat{n}/m$.
for $\tilde{\mathbf{j}} \in \mathbb{S}^m$ **do**
  Compute: $\bar{a}_{1,\tilde{j}} = \lfloor M\tilde{\mathbf{a}}_1 \cdot \tilde{\mathbf{j}} - Mb \rceil$
  for $k = 2$ to $\hat{n}/m$ **do**
    Compute: $\bar{a}_{k,\tilde{\mathbf{j}}} = \lfloor M\tilde{\mathbf{a}}_k \cdot \tilde{\mathbf{j}} \rceil$.
  end for
end for
for $k = 1$ to $\hat{n}/m$ **do**
  Initialize: $\mathrm{RGSW}_s(H_k) = \mathrm{RGSW}_s(0)$
  for $\tilde{\mathbf{j}} \in \mathbb{S}^m$ **do**
    Compute: $\mathrm{RGSW}_s(H_k) = \mathrm{RGSW}_s(H_k) \bigoplus \left(X^{\bar{a}_{k,\tilde{\mathbf{j}}}} \cdot \mathrm{RGSW}_s(\delta_{\tilde{\mathbf{j}},\tilde{\mathbf{s}}_k}) \right)$
  end for
end for
Initialize: $\mathrm{ACC}^* = (0, v^*(X))$.
for $k = 1$ to $\hat{n}/m$ **do**
  Compute: $\mathrm{ACC}^* = \mathrm{RGSW}_s(H_k) \boxdot \mathrm{ACC}^*$
end for
Output: $\mathrm{ACC}^* = \mathrm{RLWE}^*_s(X^{-\imath} v^*(X))$ with $\imath = -\sum_{k=1}^{\hat{n}/m} \bar{a}_{k,\tilde{\mathbf{s}}_k}$.

Extraction

Assuming that the polynomial key s has been constructed from the key $\mathbf{s} = (s_0, \ldots, s_{N-1})$ as

$$s(X) = \sum_{j=0}^{N-1} s_j X^j,$$

the extraction of the trace $\mathrm{Tr}(X^{-\imath} \cdot v^*(X))$ is straightforward using formula (19) of Proposition 6.1 with $i = 0$. That is to say, given the encryption $(a^*(X), b^*(X)) = \mathrm{RLWE}^*_s(X^{-\imath} \cdot v^*(X)) \in \mathcal{T}^\vee \times \mathcal{T}^\vee$ of the ACC*-output of the blind rotate, a LWE-encryption with key $\mathbf{s} = (s_0, \ldots, s_{N-1})$ of $\mathrm{Tr}(X^{-\imath} \cdot v^*(X)))$ is obtained as

$$\mathrm{LWE}_{\mathbf{s}}(\mathrm{Tr}(X^{-\imath} \cdot v^*(X))) = (\mathbf{a}, b),$$

with

$$b = b_0^* \quad \text{and} \quad \begin{cases} \mathbf{a}_1 = a_0^* & \\ \mathbf{a}_j = -\sum_{\ell=1}^{t-1} a^*_{\ell\frac{M}{t}-j+1} & \text{for } 2 \le j \le \frac{M}{t} + 1 \\ \mathbf{a}_j = a^*_{M-j+1} & \text{for } \frac{M}{t} + 2 \le j \le N. \end{cases} \tag{19}$$

7.4 Error estimate of the bootstrap output

An interesting question now arises as to whether the validity of the output error estimate, established in the standard case $t = 2$, extends to other primes within the current framework. The next proposition will demonstrate that the increase in variance when moving from $t = 2$ to $t \ge 3$ is bounded by a factor of two-a growth exactly corresponding to the variance increase resulting from a single addition.

Proposition 7.4 *Assume $\hat{\mathbf{s}} \in \mathbb{S}^{\hat{n}}$, with an integer $1 \leq m \leq \hat{n}$ dividing $\hat{n}$, and $s(X) = \sum_{j=0}^{N-1} s_j X^j$. Consider bootstrapping keys*

$$\mathrm{RGSW}_s(\delta_{\tilde{\mathbf{j}},\tilde{\mathbf{s}}_i}), \quad 1 \leq i \leq \frac{\hat{n}}{m}, \quad \tilde{\mathbf{j}} \in \mathbb{S}^m,$$

where $\tilde{\mathbf{s}}_i$ are as per (10) and $\delta_{\tilde{\mathbf{j}},\tilde{\mathbf{s}}_i}$ as per (12) and define $c_*^{(0)}$ the trivial noise-free RLWE$_s^*$ encryption of v^* and, for $k = 1, \ldots, \hat{n}/m$

$$c_*^{(k)} = (a_*^{(k)}, b_*^{(k)}) = C^{(k)} \boxdot c_*^{(k-1)} \text{ where } C^{(k)} = \bigoplus_{\tilde{\mathbf{j}} \in \mathbb{S}^m} \left(X^{\bar{a}_{k,\tilde{\mathbf{j}}}} \cdot \mathrm{RGSW}_s(\delta_{\tilde{\mathbf{j}},\tilde{\mathbf{s}}_k}) \right).$$

Then the variance $\mathrm{Var}(\mathcal{E})$ of the LWE-encryption error $\mathcal{E} = \langle 1, c_*^{(n/m)} \rangle$ of the bootstrap output is given by

$$\left(\sum_{j=0}^{\frac{\hat{n}}{m}-1} \gamma \left(\sum_{r=j+1}^{\frac{\hat{n}}{m}} \nu_r \right) \right) (1 + \|\bar{s}\|_2^2) \frac{D^{-2\ell}}{12} + \left(\sum_{j=0}^{\frac{\hat{n}}{m}-1} \Gamma \left(\bar{a}_{j+1,\tilde{\mathbf{j}}} + \sum_{r=j+2}^{k-1} \nu_r \right) \right) \ell \frac{D^2 + \xi_D}{6} |\mathbb{S}|^m \sigma_{BK}^2 \tag{20}$$

with $\xi_D = 2$ if D is even and $\xi_D = -1$ if D is odd and with $\bar{s}(X) = \sum_{i=0}^{N-1} \bar{s}_i X^i$ and $\|\bar{s}\|_2^2 = \sum_{i=0}^{N-1} \bar{s}_i^2$. The cardinal of $\mathbb{S}$ is here denoted $|\mathbb{S}|$ and γ and Γ are defined in Appendix (corollaries. 9.7,9.8).

Remark 7.5 *The following bound holds for all $t \geq 2$*

$$\forall 0 \leq \nu \leq M-1, \quad \Gamma(\nu) \leq 2N,$$

and if we suppose that ν is a random variable uniformly distributed in $\{0, \ldots, M-1\}$, then

$$\mathbb{E}(\gamma(\nu)) \leq 2.$$

This implies that, in practice, the variance of the bootstrap error for $t \geq 3$ is at most twice as large as in the standard case $t = 2$.

Proof. The following induction, valid for $1 \leq k \leq \hat{n}/m$, can be easily derived from Proposition 5.12:

$$\begin{aligned} \mathrm{Err}(c_*^{(k)}) &= X^{\nu_k} \mathrm{Err}(c_*^{(k-1)}) + X^{\nu_k} \delta_*^{(k-1)} + \Lambda_*^{(k)} \\ &= \sum_{j=0}^{k-1} \left(\prod_{r=j+2}^{k} X^{\nu_r} \right) \left(X^{\nu_{j+1}} \delta_*^{(j)} + \Lambda_*^{(j+1)} \right), \end{aligned}$$

where

$$\delta_*^{(j)} = \delta(b_*^{(j)}) - s\delta(a_*^{(j)}) \quad \text{and} \quad \Lambda_*^{(j+1)} = \mathrm{dec}_{B,\ell}(c_*^{(j)}) \ \mathrm{Err}(C^{(j+1)}).$$

The error embedded in the bootstrap output is obtained by extracting an LWE-encryption of the trace of $c_*^{(n/m)}$, i.e., $\mathcal{E} = \langle 1, c_*^{(n/m)} \rangle$. Assuming all variables are independent and centered, its variance is given by

$$\mathrm{Var}(\mathcal{E}) = \sum_{j=0}^{k-1} \mathrm{Var} \left(\left\langle 1, \left(\prod_{r=j+1}^{k} X^{\nu_r} \right) \delta_*^{(j)} \right\rangle \right) + \sum_{j=0}^{k-1} \mathrm{Var} \left(\left\langle 1, \left(\prod_{r=j+2}^{k} X^{\nu_r} \right) \Lambda_*^{(j+1)} \right\rangle \right).$$

The first sum can be split into two terms, computed using Corollary 9.7 of the Appendix:

$$\mathrm{Var}\left(\left\langle 1, \left(\prod_{r=j+1}^{k} X^{\nu_r}\right)\delta(b_*^{(j)})\right\rangle + \left\langle 1, \left(\prod_{r=j+1}^{k} X^{\nu_r}\right) s\delta(a_*^{(j)})\right\rangle\right) = \gamma\left(\sum_{r=j+1}^{k}\nu_r\right)(1+\|\bar{s}\|_2^2)\frac{D^{-2\ell}}{12}.$$

Similarly, the second sum is computed as follows (using Corollary 9.8 of the Appendix):

$$\mathrm{Var}\left(\sum_{g=1}^{\ell-1}\sum_{\tilde{j}\in\mathbb{S}^m}\left\langle 1, \left(X^{\bar{a}_{j+1,\tilde{\mathbf{j}}}}\prod_{r=j+2}^{k} X^{\nu_r}\right)\mathrm{dec}_{B,\ell}(c_*^{(j)})\ \mathrm{Err}(C^{(j+1)})\right\rangle\right) = 2\ell|\mathbb{S}|^m\Gamma_{j+1}\frac{D^2+\xi_D}{12},$$

where

$$\Gamma_{j+1} = \Gamma\left(\bar{a}_{j+1,\tilde{\mathbf{j}}} + \sum_{r=j+2}^{k-1}\nu_r\right).$$

This completes the proof.

7.5 Prime-power bootstrapping versus standard one

This subsection showcases the enhanced flexibility of prime-power bootstrapping compared to its standard counterpart. To illustrate this, we fix several parameters as follows:

- **Security parameters:** Consistent with common practice, we select binary keys $|\mathbb{S}| = 2$ and $\mathbb{S} = \{0,1\}$, aiming for 128-bit security. Using the LWE-estimator from [1]-maintained by Martin Albrecht-for $q = 2^{64}$, this translates to a noise standard deviation:
$$\sigma_{BK} = 2^{-0.0265N+1.8709},$$
where $N = \varphi(M)$ is the degree of the cyclotomic polynomial as well as the dimension of the ambiant LWE-mask.

- **Design parameters:** We choose the collapse parameter $m = 4$, balancing computational efficiency and accuracy. For the decomposition steps (see Section 5.5), we set $\ell_{KS} = 16$, $D_{KS} = 2$, $\ell_{BR} = 6$, and $D_{BR} = 2^{16}$. These parameters influence the error similarly for $t = 2$ and larger primes $t \geq 3$.

- **Messages and external products:** To examine how variations in N affect performance, we fix $\hat{n} = 464$ and $t = 31$.

The expected computational cost-neglecting the very fast extraction phase-is approximated by summing the costs of the bootstrap and key-switch operations:

$$\mathcal{C}_{BR} + \mathcal{C}_{KS},$$

where

$$\mathcal{C}_{BR} = \hat{n}/m\left(8N(K^m-1)\ell_{BR} + 4N\left(1+\frac{3}{2}\log_2 N\right) + 4N\ell_{BR}\right),$$

and

$$\mathcal{C}_{KS} = (\ell_{KS}+3)\times\frac{3}{2}N\log_2 N + N(3\ell_{KS}-1).$$

Note that for $t \geq 3$, substitute N with M in the $\mathcal{C}_{BR}$ formula; $\mathcal{C}_{KS}$ remains unchanged. Next, consider a ciphertext with error e, modeled as a centered Gaussian with standard deviation σ, with density function:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}}.$$

The probability of decryption failure-i.e., $|e| > \frac{1}{2p}$-is:

$$\mathbb{P}(\text{fail}) = \mathbb{P}\left(|e| > \frac{1}{2p}\right) = 2\int_{1/(2p)}^{\infty} f(x)dx = \operatorname{erfc}\left(\frac{1}{2p\sigma\sqrt{2}}\right).$$

Thus, for a prescribed failure probability $\mathbb{P}_{\text{fail}}$, the noise threshold is:

$$\sigma_{\text{threshold}} = \frac{1}{2p\sqrt{2}\operatorname{erfc}^{-1}(\mathbb{P}_{\text{fail}})}.$$

When multiple bootstrap outputs are combined via addition, the resulting ciphertext remains correct with high probability as long as:

$$n_{\text{add}}\,\sigma_{\mathcal{E}}^2 \leq \sigma_{\text{threshold}}^2,$$

where $\sigma_{\mathcal{E}}$ is the error standard deviation after one bootstrap. Building on this setting, we analyze the maximum number of additions n_{add} permitted before the failure probability exceeds $\mathbb{P}_{\text{fail}}$. Specifically, we plot:

$$\left\lfloor \frac{\sigma_{\text{threshold}}^2}{\sigma_{\mathcal{E}}^2} \right\rfloor$$

against $\mathcal{C}_{BR} + \mathcal{C}_{KS}$ for several N values, using the variance $\mathrm{Var}(\mathcal{E}) = \sigma_{\mathcal{E}}^2$ derived from formula (20). Here, the sums involving γ and Γ are replaced by 2 and $2N$ for $t \geq 3$, and by 1 and N for $t = 2$. We have highlighted in red the points corresponding to values of N that are powers of two, specifically $N = 2048$ and $N = 4096$. It is evident from this distinction that if we require our cryptosystem to support a specific number of additions, choosing non-power-of-two values for N provides significantly more options and flexibility.

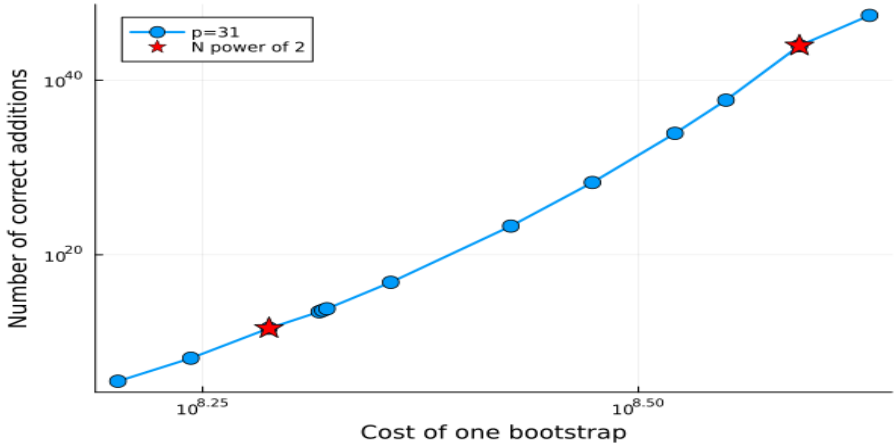

Figure 1: Number of correct additions of bootstrap outputs versus the cost of a single bootstrap.

8 Trace operators and their homomorphic evaluations

The trace operator is a crucial concept in the study of algebraic fields and structures. It fundamentally allows for the mapping of elements from a field extension back to the base field. Moreover, the trace operator is utilized to analyze and manipulate algebraic structures effectively. In this section, we will demonstrate how it can be computed efficiently and highlight its significance for fast packing algorithms.

8.1 The complete trace and its encryption

We recall that if $\mathcal{K}$ is a Galois extension of a number field $\mathcal{K}_0$, then the associated Galois group $\mathrm{Gal}(\mathcal{K}/\mathcal{K}_0)$ is defined as the collection of all automorphisms of $\mathcal{K}$ that leave the subfield $\mathcal{K}_0$ unchanged. In our context, the field $\mathcal{K} = \mathbb{Q}[X]/\Phi_M(X)$ serves as a Galois extension of the field $\mathcal{K}_0 = \mathbb{Q}$. It is well-known that the Galois group in this scenario comprises the automorphisms τ_d defined by

$$\tau_d(P)(X) = P(X^d) \quad \text{for all} \quad P \in \mathcal{K},$$

where d is any integer that is co-prime to M. Notably, this group is isomorphic to $\mathbb{Z}_M^\times$ and its cardinality is precisely $N = \varphi(M)$, which also represents the degree of the extension. In what follows, we will explicitly denote the extension and base fields in the notation of the trace (refer to Definition 3.4) as follows:

$$\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_0}(P)(X) = \sum_{\substack{1 \le d \le M \\ \gcd(d,M)=1}} P(X^d), \quad \text{for all } P \in \mathcal{K}.$$

We have the following useful identities, from which the expression of the dual basis $(\Omega_i^*)_{0 \le i < N}$ can be derived (see Formula (3)). Their proof is straightforward and therefore omitted:

Proposition 8.1 *When $M = t^\alpha$ with $\alpha \ge 1$ and t being a prime, we have the following identity in $\mathcal{K}$: For all $n, k \in \mathbb{Z}$,*

$$\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_0}(X^n) = 0 \ \text{ if } [n]_{M/t} \neq 0 \text{ and } \mathrm{Tr}_{\mathcal{K}/\mathcal{K}_0}\left(X^{kM/t}\right) = M\delta_{[k]_t,0} - \frac{M}{t}.$$

As a result, we obtain the following:

- *For any polynomial $\mu(X) = \sum_{n=0}^{N-1} \mu_n X^n \in \mathcal{K}$*

$$\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_0}(\mu(X)) = \sum_{k=0}^{t-2} \left(M\delta_{[k]_t,0} - \frac{M}{t} \right) \mu_{kM/t} = N\mu_0 - \frac{M}{t} \sum_{k=1}^{t-2} \mu_{kM/t}.$$

- *For any index $i \in \mathbb{Z}$ and any polynomial $\mu(X) = \sum_{n=0}^{N-1} \mu_n X^n \in \mathcal{K}$, we have*

$$\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_0}\left(\bar{\Omega}_i^*(X)\mu(X)\right) = \mu_i,$$

where the definition of the coefficients μ_i is extended to indices i in $\mathbb{Z}$ as above.

Now, to derive an RLWE-encryption of the quantity $\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_0}(\mu(X))$ from a RLWE-encryption of $\mu(X) \in \mathcal{T}$, a well-established and intuitive strategy can be employed. Start with an encryption $c(X) = (a(X), b(X) \in \mathcal{T} \times \mathcal{T}$ of $\mu(X)$ using the key $s(X) \in \mathcal{R}$:

$$b(X) = s(X) \cdot a(X) + \mu(X) + e(X) \bmod \mathcal{R}.$$

For any integer d that is co-prime to M, we can rewrite the expression as follows:

$$b(X^d) = s(X^d) \cdot a(X^d) + \mu(X^d) + e(X^d) \bmod \mathcal{R}.$$

Since $\Phi_M(X^d)$ is a multiple of $\Phi_M(X)$ for all integers d co-prime to M (see Proposition 3.2), it follows that $(a(X^d), b(X^d))$ constitutes an encryption of $\mu(X^d)$ with the secret key $s(X^d)$. Next, a simple key switching from $s(X^d)$ to $s(X)$ transforms this encryption into one of $\mu(X^d)$ under the key $s(X)$, which is now independent of d. Finally, by summing these encryptions over all integers d that are less than M and co-prime to M, we obtain an encryption of $\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_0}(\mu(X))$ with the key $s(X)$.

Algorithm 16 RLWE Encryption of the Trace of $\mu(X) \in \mathcal{T}$

1: **Input:** RLWE encryption $c(X) = (a(X), b(X)) \in \mathcal{T} \times \mathcal{T}$ of $\mu(X)$ with key $s(X) \in \mathcal{R}$
2: **Initialize:** $(a_\Sigma(X), b_\Sigma(X)) = (a(X), b(X)) \in \mathcal{T} \times \mathcal{T}$
3: **for** $d = 2$ **to** $M - 1$ such that $\gcd(d, M) = 1$ **do**
4: **Compute:** $(a(X^d), b(X^d)) \bmod \mathcal{R}$
5: **Perform key switching:**

$$(a_d(X), b_d(X)) = \mathrm{KeySwitch}_{s(X^d) \to s(X)} \Big((a(X^d), b(X^d)) \Big)$$

6: $(a_\Sigma(X), b_\Sigma(X)) \leftarrow (a_\Sigma(X), b_\Sigma(X)) + (a_d(X), b_d(X))$
7: **end for**
8: **Output:** $(a_\Sigma(X), b_\Sigma(X)) = \mathrm{RLWE}_{s(X)}(\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_0}(\mu(X)))$ with key $s(X)$

Despite its straightforward nature, we note that the algorithm necessitates

$$\varphi(M) = N = (t-1)t^{\alpha-1}$$

key-switches to obtain homomorphic encryptions of all quantities $P(X^d)$ from the encryption of $P(X)$. To mitigate this cost, we will utilize a Galois tower of field extensions

$$\mathbb{Q} = \mathcal{K}_0 \subset \mathcal{K}_1 \subset \ldots \subset \mathcal{K}_\alpha = \mathcal{K}$$

along with the associated partial traces.

8.2 Partial traces and fast evaluation of the complete trace

When dealing with a tower of field extensions $\mathbb{Q} \subseteq \mathcal{L} \subseteq \mathcal{K}$, the complete trace from $\mathcal{K}$ to $\mathbb{Q}$ can be efficiently expressed in terms of partial traces. The procedure involves first calculating the partial trace from $\mathcal{K}$ down to $\mathcal{L}$, and then from $\mathcal{L}$ to $\mathbb{Q}$. This decomposition effectively simplifies what might otherwise be a complex computation into more manageable steps. We will explore these optimization opportunities further in this section.

8.2.1 Fast evaluation of the trace: a first approach

A tower of fields can be intuitively constructed by introducing, for $1 \leq j \leq \alpha$, the extensions $\mathcal{K}_j$ over $\mathbb{Q}$, which have degree $(t-1)t^{j-1}$, defined as follows:

$$\begin{aligned}
\mathbb{Q} = \mathcal{K}_0 &:= \left\{ P(X^{M/1}) \bmod \Phi_M(X), P \in \mathbb{Q}[X] \right\}, \\
\mathcal{K}_1 &:= \left\{ P(X^{M/t}) \bmod \Phi_M(X), P \in \mathbb{Q}[X] \right\}, \\
\mathcal{K}_2 &= \left\{ P(X^{M/t^2}) \bmod \Phi_M(X), P \in \mathbb{Q}[X] \right\}, \\
&\vdots \\
\mathcal{K}_j &= \left\{ P(X^{M/t^j}) \bmod \Phi_M(X), P \in \mathbb{Q}[X] \right\}, \\
&\vdots \\
\mathcal{K}_\alpha &= \left\{ P(X^{M/t^\alpha}) \bmod \Phi_M(X), P \in \mathbb{Q}[X] \right\} = \mathcal{K}.
\end{aligned}$$

It is noteworthy that $\mathcal{K}_1$ is isomorphic to the field $\mathbb{Q}[X]/\Phi_t(X)$, while $\mathcal{K}_2$ corresponds to the field $\mathbb{Q}[X]/\Phi_{t^2}(X)$. This pattern continues until $\mathcal{K}_\alpha$ which coincides with $\mathcal{K} = \mathbb{Q}[X]/\Phi_M(X)$. We then examine the tower structure defined by the following inclusions:

$$\mathbb{Q} = \mathcal{K}_0 \subset \mathcal{K}_1 \subset \mathcal{K}_2 \subset \cdots \subset \mathcal{K}_{\alpha-1} \subset \mathcal{K}_\alpha = \mathcal{K},$$

which enables the decomposition of the trace utilizing the tower structure of the associated Galois groups as follows:

$$\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_0} = \mathrm{Tr}_{\mathcal{K}_1/\mathcal{K}_0} \circ \mathrm{Tr}_{\mathcal{K}_2/\mathcal{K}_1} \circ \cdots \circ \mathrm{Tr}_{\mathcal{K}/\mathcal{K}_{\alpha-1}}. \tag{21}$$

Lemme 8.2 *The Galois groups associated to the successive field extensions may be described as follows: for all* $1 \leq j \leq \alpha$*,*

$$\mathrm{Gal}(\mathcal{K}_j/\mathcal{K}_0) = \left\{ \tau_d \,|\, 0 \leq d \leq t^j - 1,\, \gcd(d,t) = 1 \right\},$$

and for all $0 \leq j \leq \alpha - 1$*:*

$$\mathrm{Gal}(\mathcal{K}/\mathcal{K}_j) = \left\{ \tau_d \,|\, d = kt^j + 1,\, 0 \leq k \leq t^{\alpha-j} - 1 \right\}.$$

Furthermore, we have, for all $1 \leq j \leq \alpha$*:*

$$\mathrm{Gal}(\mathcal{K}_{j+1}/\mathcal{K}_j) = \left\{ \tau_d \,|\, d = kt^j + 1,\, 0 \leq k \leq t - 1 \right\}. \tag{22}$$

Proof. To prove (22), observe that we must have

$$\forall\, \tau_d \in \mathrm{Gal}(\mathcal{K}_{j+1}/\mathcal{K}_j), \quad \tau_d\left(X^{M/t^j}\right) = X^{M/t^j}.$$

Hence, $X^{(d-1)M/t^j} = 1$, so that $(d-1)M/t^j = 0 \bmod M$. In other words, $d = kt^j + 1$ for some $k \in \mathbb{Z}$. In order to prove that we only need to consider values of k in $\{0, \ldots, t-1\}$, we decompose $k = qt + r$ with $0 \leq r \leq t-1$, and notice that

$$\frac{dM}{t^{j+1}} = \frac{(kt^j+1)M}{t^{j+1}} = \frac{((qt+r)t^j+1)M}{t^{j+1}} = \frac{((rt^j+1)M}{t^{j+1}} \bmod M.$$

This completes the proof of (22). **Efficiency:** We now elucidate why the decomposition formula (21) facilitates a more efficient homomorphic evaluation of an encryption of $\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_0}(\mu(X))$. The costs can be assessed in terms of the number of necessary automorphisms (or key-switchings). Recall that directly computing the complete trace requires $N-1=(t-1)t^{\alpha-1}-1$ non-trivial automorphisms. By utilizing the decomposition (21), only $t-1$ non-trivial automorphisms are needed for each partial trace $\mathrm{Tr}_{\mathcal{K}_{j+1}/\mathcal{K}_j}$, for $1 \leq j \leq \alpha-1$, and $t-2$ automorphisms for evaluating $\mathrm{Tr}_{\mathcal{K}_1/\mathcal{K}_0}$. Indeed, the order of the Galois group $\mathrm{Gal}_{\mathcal{K}_{j+1}/\mathcal{K}_j}$ is t for $1 \leq j \leq \alpha-1$, and $t-1$ for $j=0$. Consequently, the total number of required automorphisms is $(\alpha-1)(t-1)+t-2=\alpha(t-1)-1$, which is significantly less than $N-1$.

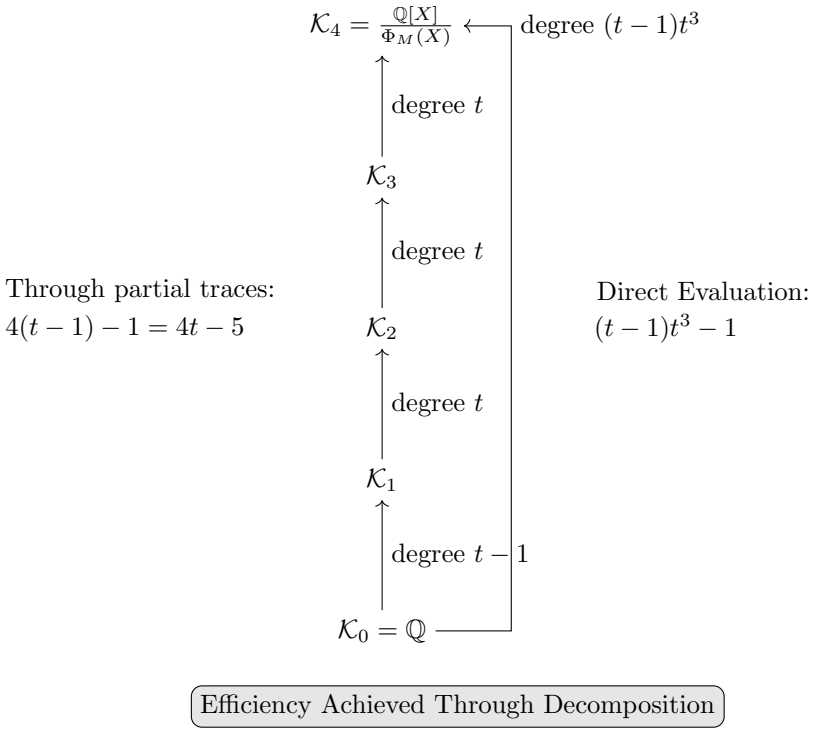

Figure 2: Illustration of the Galois tower with $\alpha=4$ and the decomposition of the trace.

The structure of the Galois groups allows us to express the successive traces, for $1 \leq j \leq \alpha-1$, as

$$\forall\, P \in \mathcal{K},\ \mathrm{Tr}_j(P)(X) = \sum_{0 \leq k \leq t-1} P(X^{kt^j+1}), \tag{23}$$

and for the case $j=0$:

$$\forall\, P \in \mathcal{K},\ \mathrm{Tr}_0(P)(X) = \sum_{0 \leq k \leq t-1} P(X^k). \tag{24}$$

Note that $\mathrm{Tr}_j(P)(X) = \mathrm{Tr}_{\mathcal{K}_{j+1}/\mathcal{K}_j}(P)(X)$ for $P \in \mathcal{K}_{j+1}$. Introducing the notation for $0 \leq j \leq \alpha$ (note that $\mathcal{T}_\alpha = \mathcal{T}$ and $\mathcal{T}_0 \equiv \mathbb{T}$)

$$\mathcal{T}_j = \left\{ P(X^{M/t^j}) \bmod \mathbb{Z}[X] \bmod \Phi_M(X), P \in \mathbb{Q}[X] \right\},$$

their homomorphic computation can be carried out as described in the following algorithm:

Algorithm 17 Partial Trace $\mathrm{Tr}_j(\mu)$ of $\mu(X) \in \mathcal{K}$ for $1 \leq j \leq \alpha - 1$

Input: RLWE encryption $c(X) = (a(X), b(X)) \in \mathcal{T} \times \mathcal{T}$ of $\mu(X)$ with key $s(X) \in \mathcal{R}$
Initialize: $(a_\Sigma(X), b_\Sigma(X)) = (a(X), b(X)) \in \mathcal{T} \times \mathcal{T}$
for $k = 1$ **to** $t - 1$ **do**
  Compute: $d = kt^j + 1$
  Compute: $(a(X^d), b(X^d)) \bmod \mathcal{R}$
  Perform key switching: $(a_d(X), b_d(X)) = \mathrm{KeySwitch}_{s(X^d) \to s(X)} \left((a(X^d), b(X^d))\right)$
  $(a_\Sigma(X), b_\Sigma(X)) \leftarrow (a_\Sigma(X), b_\Sigma(X)) + (a_d(X), b_d(X))$
end for
Output: $(a_\Sigma(X), b_\Sigma(X)) = \mathrm{RLWE}_{s(X)}(\mathrm{Tr}_{\mathcal{K}_{j+1}/\mathcal{K}_j}(\mu(X)))$ with key $s(X)$

We will temporarily defer discussion of a similar algorithm for the trace Tr_0 since it can be further decomposed and optimized. In the meantime, we can present the following:

Proposition 8.3 *Let $M = t^\alpha$ with $\alpha \geq 1$, where t is a prime. We have the following identities in $\mathcal{K}$: For all $\nu \in \mathbb{Z}$ and all $1 \leq j \leq \alpha - 1$, the trace is given by:*

$$\mathrm{Tr}_j\left(X^\nu\right) = \begin{cases} tX^\nu & \text{if } [\nu]_{M/t^j} = 0, \\ 0 & \text{if } [\nu]_{M/t^j} \neq 0 \quad \text{and} \quad [\nu]_{M/t^{j+1}} = 0, \\ \sum_{k=0}^{t-1} X^{\nu\left(1+kt^j\right)} & \text{if } [\nu]_{M/t^{j+1}} \neq 0. \end{cases}$$

And for $j = 0$:

$$\mathrm{Tr}_0\left(X^\nu\right) = \begin{cases} t-1 & \text{if } [\nu]_M = 0, \\ -1 & \text{if } [\nu]_M \neq 0 \quad \text{and} \quad [\nu]_{M/t} = 0, \\ \sum_{k=1}^{t-1} X^{\nu k} & \text{if } [\nu]_{M/t} \neq 0. \end{cases}$$

In particular

$$\mathrm{Tr}_0\left((1 - X^{M/t})X^\nu\right) = \begin{cases} t & \text{if } [\nu]_M = 0, \\ 0 & \text{if } [\nu]_M \neq 0 \quad \text{and} \quad [\nu]_{M/t} = 0, \\ \sum_{k=1}^{t-1} X^{\nu k}\left(1 - X^{kM/t}\right) & \text{if } [\nu]_{M/t} \neq 0. \end{cases}$$

8.2.2 Fast evaluation of the Trace: the algebraic approach

We recall that a tower of fields can be constructed using the corresponding tower of Galois groups via the fundamental theorem of Galois theory. This theorem asserts that for every subgroup G of $\mathrm{Gal}(\mathcal{K}/\mathbb{Q})$, there exists an intermediate field $\mathcal{G}$ such that $\mathbb{Q} \subseteq \mathcal{G} \subseteq \mathcal{K}$, with $\mathcal{G}$ being the fixed field of G. Specifically, the fixed field of G comprises those elements in $\mathcal{K}$ that remain unchanged under all automorphisms in G. In our context, identifying subgroups of $\mathrm{Gal}(\mathcal{K}/\mathbb{Q})$ is particularly straightforward, as these subgroups are isomorphic to the subgroups of $\mathbb{Z}_M^\times$. To enumerate the elements of these Galois subgroups, it is customary to utilize the generators of the group $\mathbb{Z}_M^\times$. It is well-known that for a prime $t \geq 3$, the group $(\mathbb{Z}_{t^\alpha})^\times$ has a generator, unlike the situation for $t = 2$, which only possesses a generator if $\alpha = 2$ or $\alpha = 4$. Furthermore, this generator can be readily derived in one of the following ways:

- If $\tilde{g}$ is of order $t - 1$ in $(\mathbb{Z}_{t^\alpha})^\times$ then $g = (t + 1)\tilde{g}$ serves as a generator of $(\mathbb{Z}_{t^\alpha})^\times$;
- Alternatively, if $\tilde{g}$ is a generator of $\mathbb{Z}_t^\times$, it is known that either $g = \tilde{g}$ or $g = \tilde{g} + t$ will act as a generator of $\mathbb{Z}_{t^2}^\times$ and also a generator of $(\mathbb{Z}_{t^\beta})^\times$ for any power $\beta \geq 2$.

Thus, assuming that g is a generator of $(\mathbb{Z}_{t^\alpha})^\times$ coinciding with $\tilde{g}$ or $\tilde{g}+t$ (where $\tilde{g}$ is a generator of $\mathbb{Z}_t^\times$), we have

$$\mathrm{Gal}(\mathcal{K}/\mathbb{Q}) = \{\tau_{g^k},\ k = 0, \ldots, N-1\} \cong \mathbb{Z}_M^\times \cong (\mathbb{Z}_N, +)$$

and we have the following sequence of inclusions of additive sub-groups:

$$(t-1)t^{\alpha-1}\mathbb{Z}_1 \subset (t-1)t^{\alpha-2}\mathbb{Z}_t \subset (t-1)t^{\alpha-3}\mathbb{Z}_{t^2} \subset \ldots \subset (t-1)t\mathbb{Z}_{t^{\alpha-2}} \subset (t-1)\mathbb{Z}_{t^{\alpha-1}} \subset \mathbb{Z}_N,$$

to which we can associate a tower of Galois groups

$$\{\tau_1\} \subset G_1 = \{\tau_{g^k},\ k \in (t-1)t^{\alpha-2}\mathbb{Z}_t\} \subset \ldots \subset G_{\alpha-1} = \{\tau_{g^k},\ k \in (t-1)\mathbb{Z}_{t^{\alpha-1}}\} \subset \mathrm{Gal}_{\mathcal{K}/\mathbb{Q}},$$

and by the Galois correspondence, a tower of Galois fields

$$\mathcal{K} \supset \mathcal{K}_{\alpha-1} \supset \ldots \supset \mathcal{K}_1 \supset \mathbb{Q}. \tag{25}$$

Note that the fixed field $\mathcal{G}_j$ associated with G_j for $1 \leq j \leq \alpha-1$ is unique and must coincide with with $\mathcal{K}_j$ due to the following relationships:

$$[\mathcal{K} : \mathcal{K}_j] = |\mathrm{Gal}(\mathcal{K}/\mathcal{K}_j)| = t^{\alpha-j} = |G_j| = [\mathcal{K} : \mathcal{G}_j].$$

In summary, we can state the following:

Lemme 8.4 *The successive Galois groups associated with the field tower described in (25) can be characterized as follows:*

$$\mathrm{Gal}(\mathcal{K}_{j+1}/\mathcal{K}_j) = \left\{\tau_d,\ d = g^{k(t-1)t^{j-1}},\ 0 \leq k \leq t-1\right\}, \quad \textit{for all } 1 \leq j \leq \alpha-1,$$
$$\mathrm{Gal}(\mathcal{K}_j/\mathcal{K}_0) = \left\{\tau_d, d = g^k,\ 0 \leq k \leq (t-1)t^{j-1}-1\right\}, \quad \textit{for all } 1 \leq j \leq \alpha.$$

The corresponding composition of partial traces

$$\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_0} = \mathrm{Tr}_0 \circ \mathrm{Tr}_1 \circ \cdots \circ \mathrm{Tr}_{\alpha-1}.$$

involves

$$(\alpha-1)(t-1) + t - 2 = \alpha(t-1) - 1$$

non-trivial automorphisms.

Another "natural" tower of Galois groups is given by the sequence

$$\{\tau_1\} \subset H_1 = \{\tau_{g^k},\ k \in t^{\alpha-1}\mathbb{Z}_{t-1}\} \subset \ldots \subset H_{\alpha-1} = \{\tau_{g^k},\ k \in t\mathbb{Z}_{(t-1)t^{\alpha-2}}\} \subset \mathrm{Gal}_{\mathcal{K}/\mathbb{Q}}$$

which is associated with the field tower

$$\mathcal{K} \supset \mathcal{H}_{\alpha-1} = \mathbb{Q}(\tau(X), \tau \in H_{\alpha-1}) \supset \ldots \supset \mathcal{H}_1 = \mathbb{Q}(\tau(X), \tau \in H_1) \supset \mathbb{Q}.$$

where $\mathbb{Q}(\tau(X), \tau \in H_j)$ represents the field generated by the elements $\tau(X), \tau \in H_j$. The associated trace decomposition differs slightly from the previous one but does not provide any computational advantage; thus, it will not be further discussed.

Remark 8.5 *In general, there are several possible configurations for towers of Galois groups. For illustration, we will comprehensively construct these towers with $t = 7$ and $\alpha = 2$. Specifically, we create the Tower of Subgroups of the Multiplicative Group $\mathbb{Z}_{49}^\times$:*

(i) The group $\mathbb{Z}_{49}^\times$ consists of integers from 1 to 48 that are coprime to 49. There are $|\mathbb{Z}_{49}^\times| = \varphi(49) = 7 \cdot 6 = 42$ elements in $\mathbb{Z}_{49}^\times$:

$$\mathbb{Z}_{49}^\times = \{1, \ldots, 6, 8, \ldots, 13, 15, \ldots, 20, 22, \ldots, 27, 29, \ldots, 34, 36, \ldots, 41, 43, \ldots, 48\}.$$

(ii) The possible orders of the subgroups must divide 42 (the order of the group). Thus, the possible orders are: $1, 2, 3, 6, 7, 14, 21, 42$.

(iii) The subgroups of $\mathbb{Z}_{49}^\times$ are thus:
Order 1: $G_0 = H_0 = \{1\}$.
Order 2: $H_{0,2} = \{1, 48\}$.
Order 3: $H_{0,3} = \{1, 18, 30\}$.
Order 6: $H_1 = \{1, 18, 19, 30, 31, 48\}$.
Order 7: $G_1 = \{1, 8, 15, 22, 29, 36, 43\}$.
Order 14: $G_{1,3} = \{1, 6, 8, 13, 15, 20, 22, 27, 29, 34, 36, 41, 43, 48\}$.
Order 21: $G_{1,2} = \{1, 2, 4, 8, 9, 11, 15, 16, 18, 22, 23, 25, 29, 30, 32, 36, 37, 39, 43, 44, 46\}$.
Order 42: $G_2 = H_2 = \mathbb{Z}_{49}^\times$.

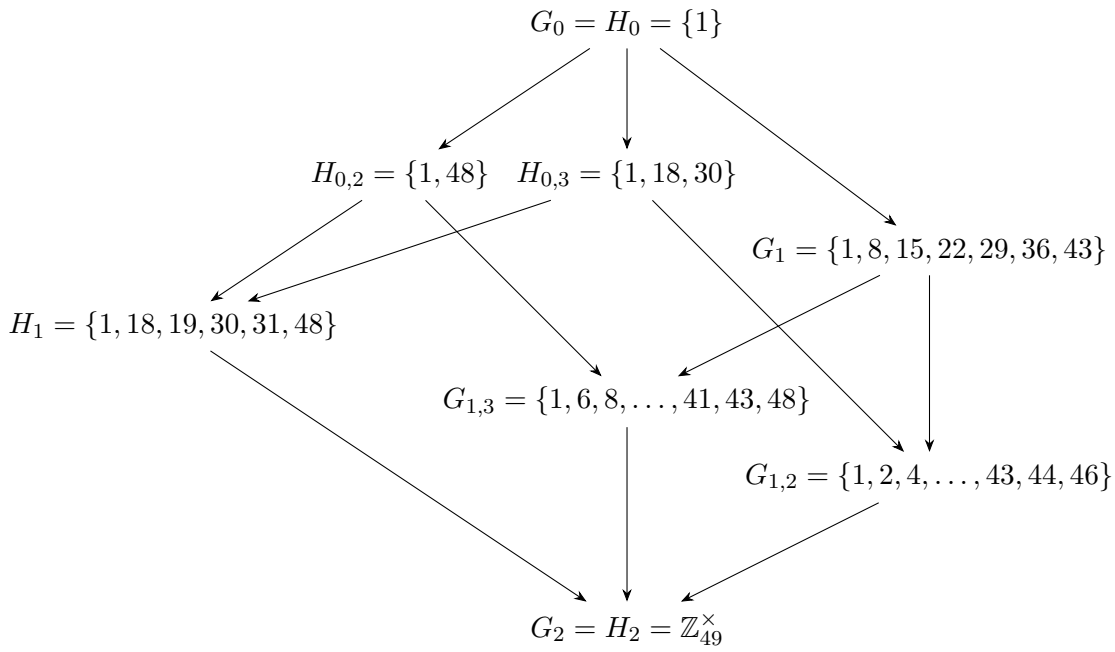

Figure 3: Tower of subgroups of the multiplicative group $\mathbb{Z}_{49}^\times$.

Note that we have not represented the more elementary towers with fewer subgroups such as $G_0 \subset H_1 \subset H_2$.

Now, a last observation is in order: as can be seen on the example in Remark 8.5, the subgroups $G_{1,2}$ or $G_{1,3}$ can be introduced in the sequence

$$G_1 \subset G_{1,2} \subset G_2 \quad \text{or} \quad G_1 \subset G_{1,3} \subset G_2.$$

If ℓ divides $t-1$, it is in fact always possible to introduce a subgroup $G_{\alpha-1,\ell}$ in the sequence of inclusions

$$G_{\alpha-1} \subset G_{\alpha-1,\ell} \subset G_\alpha = \mathbb{Z}_M^\times$$

into strict subgroups of cardinals ℓ and $(t-1)/\ell$ as soon as $t \geq 3$. The Galois subgroup $G_{\alpha-1,\ell}$ is again characterized by the corresponding subgroup $\ell\mathbb{Z}_{\frac{N}{\ell}}$ of $\mathbb{Z}_N$

$$G_{\alpha-1,\ell} = \{\tau_{g^k}, k \in \ell\mathbb{Z}_{\frac{N}{\ell}}\}$$

to which we can associated the field extension $\mathcal{K}_{1,\ell} := \mathbb{Q}(\tau(X), \tau \in G_{\alpha-1,\ell})$ in such a way that

$$\mathcal{K}_1 \supset \mathcal{K}_{1,\ell} \supset \mathbb{Q}.$$

More generally, if $t-1 = \ell_1\ell_2\cdots\ell_r$ with $r \geq 2$, we can establish the following tower of subgroups:

$$G_{\alpha-1} \subset G_{\alpha-1,\ell_1\cdots\ell_r} \subset G_{\alpha-1,\ell_1\cdots\ell_{r-1}} \subset \ldots \subset G_{\alpha-1,\ell_1} \subset G_\alpha = \mathbb{Z}_M^\times$$

Let $\mathcal{K}_{1,\ell_1\cdots\ell_j}$ denote the associated fixed field such that:

$$\mathcal{K}_1 \supset \mathcal{K}_{1,\ell_1\cdots\ell_r} \supset \mathcal{K}_{1,\ell_1\cdots\ell_{r-1}} \supset \ldots \supset \mathcal{K}_{1,\ell_1} \supset \mathbb{Q}.$$

For $2 \leq j \leq r$, we have:

$$\mathrm{Gal}(\mathcal{K}/\mathcal{K}_{1,\ell_1\ell_2\cdots\ell_{j-1}}) = \{\tau_{g^k}, k \in (\ell_1\cdots\ell_{j-1})\mathbb{Z}_{\ell_j\cdots\ell_r}\}.$$

Additionally,

$$\mathrm{Gal}(\mathcal{K}_{1,\ell_1\ell_2\cdots\ell_j}/\mathcal{K}_{1,\ell_1\ell_2\cdots\ell_{j-1}}) = \left\{\tau_{g^{k\ell_1\cdots\ell_{j-1}}},\ k \in \mathbb{Z}_{\ell_j}\right\}.$$

As an immediate consequence, the first term of the decomposition in (21) can be factored as follows:

$$\mathrm{Tr}_{\mathcal{K}_1/\mathbb{Q}} = \mathrm{Tr}_{\mathcal{K}_{1,\ell_1}/\mathbb{Q}} \circ \mathrm{Tr}_{\mathcal{K}_{1,\ell_1\ell_2}/\mathcal{K}_{1,\ell_1}} \circ \cdots \circ \mathrm{Tr}_{\mathcal{K}_{1,\ell_1\ell_2\cdots\ell_r}/\mathcal{K}_{1,\ell_1\ell_2\cdots\ell_{r-1}}}.$$

The evaluation of this trace requires:

$$\sum_{j=1}^{r}(\ell_j - 1)$$

non-trivial automorphisms instead of $t-2$. This reduction is particularly significant when $t-1 = 2^\gamma$, since

$$\sum_{j=1}^{r}(\ell_j - 1) = \sum_{j=1}^{\gamma}(2-1) = \gamma = \log_2(t-1).$$

It has become quite straightforward to obtain the following algorithm for the optimized partial trace $\mathrm{Tr}_{\mathcal{K}_1/\mathbb{Q}}$:

Algorithm 18 Partial Trace $\mathrm{Tr}_0(\mu)$ of $\mu(X) \in \mathcal{K}$

Input: $c(X) = (a(X), b(X)) = \mathrm{RLWE}_{s(X)}(\mu(X))$ of $\mu(X)$ with key $s(X) \in \mathcal{R}$, g a generator of $\mathbb{Z}_M^\times$ and $t-1$.
Initialize: $\Pi = t-1$, $\ell = 2$ and $(\tilde{a}(X), \tilde{b}(X)) = (a(X), b(X)) \in \mathcal{T} \times \mathcal{T}$
while $\Pi > 1$ **do**
  while $\ell \nmid \Pi$ **do**
    $\ell \leftarrow \mathrm{NextPrime}(\ell)$
  end while
  $\Pi \leftarrow \Pi/\ell$
  $(a_\Sigma(X), b_\Sigma(X)) \leftarrow (\tilde{a}(X), \tilde{b}(X))$
  for $k = 1$ **to** $\ell - 1$ **do**
    Compute: $d = g^{k\Pi}$
    Compute: $(\tilde{a}(X^d), \tilde{b}(X^d)) \bmod \mathcal{R}$
    Perform key switching:

$$(\tilde{a}_d(X), \tilde{b}_d(X)) = \mathrm{KeySwitch}_{s(X^d)\to s(X)}\left((\tilde{a}(X^d), \tilde{b}(X^d))\right)$$

    $(a_\Sigma(X), b_\Sigma(X)) \leftarrow (a_\Sigma(X), b_\Sigma(X)) + (\tilde{a}_d(X), \tilde{b}_d(X))$
  end for
  $(\tilde{a}(X), \tilde{b}(X)) \leftarrow (a_\Sigma(X), b_\Sigma(X))$
end while
Output: $(\tilde{a}(X), \tilde{b}(X)) = \mathrm{RLWE}_{s(X)}(\mathrm{Tr}_0(\mu(X)))$ with key $s(X)$

8.2.3 General expression of the trace for $t \geq 3$.

Assume g that is a generator of $\mathbb{Z}_M^\times$ and that $N = \ell_1 \ell_2 \cdots \ell_r$, where $(\ell_k)_{1\leq k\leq r}$ represents a sequence of prime divisors, some of which may occur more than once. We define

$$F_\ell = \{\tau_{g^k}, k \in \ell\mathbb{Z}_{\frac{N}{\ell}}\} \subset \mathbb{Z}_M^\times$$

resulting in the following sequence of subgroup inclusions:

$$\{1\} = F_{\ell_1\cdots\ell_r} \subset F_{\ell_1\cdots\ell_{r-1}} \subset \ldots \subset F_{\ell_1} \subset F_1 = \mathbb{Z}_M^\times$$

Let $\mathcal{F}_\ell$ denote the fixed field associated with F_ℓ, such that:

$$\mathcal{K} = \mathcal{F}_N = \mathcal{F}_{\ell_1\cdots\ell_r} \supset \mathcal{F}_{\ell_1\cdots\ell_{r-1}} \supset \ldots \supset \mathcal{F}_{\ell_1} \supset \mathcal{F}_1 = \mathbb{Q}$$

Consequently, the trace can be expressed as:

$$\mathrm{Tr}_{\mathcal{K}/\mathbb{Q}} = \mathrm{Tr}_{\mathcal{F}_{\ell_1}/\mathbb{Q}} \circ \mathrm{Tr}_{\mathcal{F}_{\ell_1\ell_2}/\mathcal{F}_{\ell_1}} \circ \cdots \circ \mathrm{Tr}_{\mathcal{K}/\mathcal{F}_{\ell_1\ell_2\cdots\ell_{r-1}}}.$$

The corresponding Galois groups are given for $0 \leq j \leq r$ by

$$\mathrm{Gal}(\mathcal{F}_{\ell_1\cdots\ell_j}/\mathcal{F}_{\ell_1\cdots\ell_{j-1}}) = \{\tau_{g^k}, k \in (\ell_1\cdots\ell_{j-1})\mathbb{Z}_{\ell_j}\}$$

where we adopt the convention that $\ell_0 = 1$. We now present the corresponding algorithm:

Algorithm 19 CompleteTrace: computes an encryption of $\mathrm{Tr}_{\mathcal{K}/\mathbb{Q}}(\mu)$ for $\mu(X) \in \mathcal{T}$

Input: $c(X) = (a(X), b(X)) = \mathrm{RLWE}_{s(X)}(\mu(X))$ of $\mu(X)$ with key $s(X) \in \mathcal{R}$, g a generator of $\mathbb{Z}_M^\times$ and $N = \varphi(M)$.
Initialize: $\Pi \leftarrow N$, $\ell \leftarrow 2$ and $(\tilde{a}(X), \tilde{b}(X)) \leftarrow (a(X), b(X)) \in \mathcal{T} \times \mathcal{T}$
while $\Pi > 1$ **do**
  while $\ell \nmid \Pi$ **do**
    $\ell \leftarrow \mathrm{NextPrime}(\ell)$
  end while
  $\Pi \leftarrow \Pi/\ell$
  $(a_\Sigma(X), b_\Sigma(X)) \leftarrow (\tilde{a}(X), \tilde{b}(X))$
  for $k = 1$ **to** $\ell - 1$ **do**
    Compute: $d = g^{k\Pi}$
    Compute: $(\tilde{a}(X^d), \tilde{b}(X^d)) \bmod \mathcal{R}$
    Perform key switching: $(\tilde{a}_d(X), \tilde{b}_d(X)) = \mathrm{KeySwitch}_{s(X^d)\to s(X)}\left((\tilde{a}(X^d), \tilde{b}(X^d))\right)$
    $(a_\Sigma(X), b_\Sigma(X)) \leftarrow (a_\Sigma(X), b_\Sigma(X)) + (\tilde{a}_d(X), \tilde{b}_d(X))$
  end for
  $(\tilde{a}(X), \tilde{b}(X)) \leftarrow (a_\Sigma(X), b_\Sigma(X))$
end while
Output: $(\tilde{a}(X), \tilde{b}(X)) = \mathrm{RLWE}_{s(X)}(\mathrm{Tr}_{\mathcal{K}/\mathbb{Q}}(\mu(X)))$ with key $s(X)$

We conclude this section by presenting Algorithm 20 for the partial trace $\mathrm{Tr}_{i,j}$, which maps a $\mathbb{Q}$-extension of degree i to a $\mathbb{Q}$-extension of degree $j|i$ within the field $\mathcal{K}$:

$$\mathrm{Tr}_{i,j} = \mathrm{Tr}_j \circ \ldots \circ \mathrm{Tr}_i.$$

Note that if i/j has a non-trivial divisor, say d for instance, then it is more efficient to

Algorithm 20 PartialTrace$_{i\to j}$: computes an encryption of $\mathrm{Tr}_{i,j}(\mu)$ for $\mu(X) \in \mathcal{T}$

Input: $c(X) = (a(X), b(X)) = \mathrm{RLWE}_{s(X)}(\mu(X))$ of $\mu(X)$ with key $s(X) \in \mathcal{R}$, g a generator of $\mathbb{Z}_M^\times$, $j|i$, $i|N = \varphi(M)$.
Initialize: $\ell \leftarrow i/j$ and $(a_\Sigma(X), b_\Sigma(X)) \leftarrow (a(X), b(X)) \in \mathcal{T} \times \mathcal{T}$
for $k = 1$ **to** $\ell - 1$ **do**
  Compute: $d = g^{kj}$ and $(a(X^d), b(X^d)) \bmod \mathcal{R}$
  Perform key switching: $(a_d(X), b_d(X)) = \mathrm{KeySwitch}_{s(X^d)\to s(X)}\left((a(X^d), b(X^d))\right)$
  $(a_\Sigma(X), b_\Sigma(X)) \leftarrow (a_\Sigma(X), b_\Sigma(X)) + (a_d(X), b_d(X))$
end for
Output: $(a_\Sigma(X), b_\Sigma(X)) = \mathrm{RLWE}_{s(X)}(\mathrm{Tr}_{i,j}(\mu(X)))$ with key $s(X)$

compute $\mathrm{Tr}_{i,j}$ by applying Algorithm 20 twice, specifically as $\mathrm{Tr}_{i,dj} \circ \mathrm{Tr}_{dj,j}$.

9 Fast packing operations

A complete packing operation typically processes several LWE-ciphertexts of messages $\mu_i \in \mathcal{T}$ and produces a single RLWE-ciphertext that encrypts a polynomial comprised of the μ_i's (or linear functions of the μ_i's). This procedure has been established within the framework of power of two cyclotomic polynomials; however, it remains unaddressed for general cyclotomic fields.

In this section, our objective is to generalize this construction to cyclotomic polynomials with index $M = t^\alpha$, where t is an arbitrary prime number. To achieve this, we will first demonstrate how to derive a RLWE encryption from a LWE-ciphertext containing a single message $\mu \in \mathbb{T}$, and subsequently extend this methodology to accommodate multiple LWE ciphertexts.

9.1 From a single LWE-ciphertext to an RLWE-ciphertext

Consider a LWE-ciphertext $\mathbf{c} = (\mathbf{a}, b) \in \mathbb{T}^n \times \mathbb{T}$ representing a message $\mu \in \mathbb{T}$ with the key $\mathbf{s} = (s_0, s_1, \ldots, s_{n-1})$:

$$b = \mathbf{s} \cdot \mathbf{a} + \mu + e \mod 1.$$

Our aim is to transform $\mathbf{c}$ into an RLWE encryption of $\mu \in \mathcal{T}$, treating it as a constant polynomial. In the conventional scenario where $t = 2$, it is well-established, as detailed for instance in [12], that this can be achieved through the following procedure. First, we define (assuming that $N \geq n$)

$$a(X) = \sum_{i=0}^{n-1} a_{i+1} X^{-i} \bmod (X^N + 1) \mod 1, \quad b(X) = b,$$

and

$$s(X) = \sum_{i=0}^{n-1} s_{i+1} X^i,$$

and then compute

$$\mu(X) = b - s(X) \cdot a(X).$$

The pair $(a(X), b(X))$ does not directly serve as an RLWE-encryption of μ; rather, it is an RLWE-encryption of $\mu(X)$, which encapsulates μ (with some associated noise) in its constant term. To "purify" this ciphertext and eliminate unwanted coefficients, the trace is then applied homomorphically, yielding the desired RLWE-encryption of μ.

In the broader context where $M = t^\alpha$ with t being a prime greater than 3, this methodology necessitates some modifications. We begin by (re-)defining:

$$a(X) = \sum_{i=0}^{n-1} a_{i+1} \widetilde{\Omega}_i(X) \bmod \mathcal{R}, \quad s(X) = \sum_{i=0}^{n-1} s_{i+1} X^i, \quad b(X) = b,$$

where

$$\widetilde{\Omega}_i(X) = (\bar{\Omega}_0^*)^{-1}(X) \bar{\Omega}_i^*(X) \bmod \Phi_M.$$

Note that this formulation encompasses the case $t = 2$, since in that scenario, $\widetilde{\Omega}_i(X) = X^{-i}$. Now, we define

$$\mu(X) = b(X) - s(X) \cdot a(X),$$

and we can observe from **Proposition 8.1** that the coefficient $(\mu(X))_0$ can be extracted using the following trace formula in $\mathcal{K}$:

$$\mu(X)_0 = \mathrm{Tr}\Big(\bar{\Omega}_0^*(X) \cdot \mu(X)\Big) = b - \sum_{i=1}^{n} s_i a_i = \mu + e. \tag{26}$$

Indeed, from Proposition 8.1, we have:

$$\mathrm{Tr}\left(\bar{\Omega}_0^*(X)\left(b(X) - s(X)\cdot a(X)\right)\right) = b - \sum_{i,j=0}^{n-1} s_{i+1}a_{j+1}\mathrm{Tr}\left(\bar{\Omega}_i^*(X)X^j\right) = b - \sum_{i,j=1}^{n} s_i a_j \delta_{i,j}.$$

At this juncture, several points merit attention. Firstly, it is essential to acknowledge the importance of the multiplicative kernel $\bar{\Omega}_0^*$; in its absence, the trace would result in a complex expression encompassing all coefficients of μ (refer to Proposition 8.1 for further details). Secondly, the product $\bar{\Omega}_0^*(X)\cdot\mu(X)$, which maps $\mathcal{R}^\vee \times \mathcal{T}$ to $\mathcal{T}^\vee$, along with similar products, must be calculated modulo $\mathcal{R}^\vee$ (see Remark 3.8). Therefore, the most judicious strategy is to carry out the computation

$$\mathrm{Tr}\left(\bar{\Omega}_0^*(X)\cdot\mu(X)\right)$$

in $\mathcal{K}$ and subsequently reduce the coefficients modulo 1 to return to the torus. The second step now involves evaluating the quantity

$$\mathrm{Tr}\left(\bar{\Omega}_0^*(X)\cdot\mu(X)\right)$$

homomorphically in $\mathcal{K}$. This trace is computed homomorphically through an appropri-

Algorithm 21 LWE-to-RLWE : re-encryption of a single LWE-ciphertext

Input: $\mathbf{c} = (\mathbf{a}, b) = \mathrm{LWE}_{\mathbf{s}}(\mu)$ with $b - \mathbf{s}\cdot\mathbf{a} = \mu + e$

1: $a(X) \leftarrow \sum_{i=0}^{n-1} a_{i+1}\widetilde{\Omega}_i(X) \bmod \mathcal{R}, \quad b(X) \leftarrow b.$

2: **Return** $\hat{c}(X) = (\hat{a}(X), \hat{b}(X)) = \mathrm{Tr}_0 \circ \mathrm{Tr}_1 \circ \cdots \circ \mathrm{Tr}_{\alpha-1}(\bar{\Omega}_0^*(X)a(X), \bar{\Omega}_0^*(X)b(X)).$

ate decomposition along a tower of extensions (and the corresponding Galois groups of automorphisms), as previously described:

$$\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_0} = \mathrm{Tr}_0 \circ \mathrm{Tr}_1 \circ \cdots \circ \mathrm{Tr}_{\alpha-1}. \tag{27}$$

This can be accomplished very efficiently, as demonstrated in Lemma 8.4, or even more effectively using Algorithm 19. A comprehensive description of the re-encryption procedure is provided in Algorithm 21.

9.2 Packing a set of LWE-ciphertexts into one RLWE-ciphertext

We now turn our attention to the situation in which we seek to pack $t^\beta \frac{N}{M}$ (with $1 \le \beta \le \alpha - 1$) LWE-encryptions of messages $(\mu_i)_{0\le i\le t^\beta \frac{N}{M}-1}$ into a single RLWE-encryption of

$$\sum_{i=0}^{t^\beta \frac{N}{M}-1} \mu_i X^{i\frac{M}{t^\beta}}.$$

To each μ_i, we can associate a ciphertext as previously described:

$$c^{(i)}(X) = (a^{(i)}(X), b^{(i)}(X)),$$

where the phase $\mu^{(i)}(X)$ includes μ_i as its constant coefficient (along with a certain level of noise) while the other coefficients are yet to be eliminated. Our aim is to leverage the decomposition property of the complete trace to minimize the computational cost of the

elimination process while packing the various LWE-ciphertexts together. There are three methods to achieve an RLWE-encryption of

$$\sum_{i=0}^{t^\beta \frac{N}{M}-1} \mu_i X^{i\frac{M}{t^\beta}}$$

from the ciphertexts $c^{(i)}(X)$, which we will now describe.

9.2.1 A first strategy

This first approach is somewhat naive; it involves cleaning each $c^{(i)}(X)$ by applying the complete trace operator homomorphically to each $c^{(i)}(X)$ in order to obtain a new ciphertext

$$\hat{c}^{(i)}(X) = \mathrm{RLWE}_{s(X)}\left(\mathrm{Tr}\left(\bar{\Omega}_0^*(X)\cdot \mu^{(i)}(X)\right)\right)$$

as described in the previous section, that now serves as an RLWE-encryption of μ_i. Subsequently, the expression

$$\sum_{i=0}^{t^\beta \frac{N}{M}-1} X^{i\frac{M}{t^\beta}} \cdot \hat{c}^{(i)}(X)$$

naturally provides a RLWE-encryption of

$$\sum_{i=0}^{t^\beta \frac{N}{M}-1} \mu_i X^{i\frac{M}{t^\beta}}.$$

While we can employ a rapid evaluation of the trace $\mathrm{Tr} = \mathrm{Tr}_{\mathcal{K}/\mathcal{K}_0}$, this approach remains expensive in practice, as it requires the computation of

$$t^\beta \frac{N}{M}\left((\alpha-1)(t-1)+\sum_{i=1}^{r}(\ell_j-1)\right)$$

automorphisms homomorphically (noting that $t-1=\prod_{i=1}^r \ell_r$ in this context).

Algorithm 22 LWEs-to-RLWE : packs several LWEs in an encrypted polynomial

Input: $\mathbf{c}_i = (\mathbf{a}_i, b_i) = \mathrm{LWE}_{\mathbf{s}}(\mu_i)$ for $0 \le i < t^\beta \frac{N}{M}$

1: **for** $i = 0$ to $t^\beta \frac{N}{M} - 1$ **do**
2: $\quad \hat{c}^{(i)}(X) \leftarrow$ LWE-to-RLWE$(\mathbf{a}_i, b_i)$
3: **end for**
4: **Return** $\hat{c}(X) = \sum_{i=0}^{t^\beta \frac{N}{M}-1} X^{i\frac{M}{t^\beta}} \cdot \hat{c}^{(i)}(X)$

Note: The LWE-to-RLWE *in step 2 refers to the homomorphic computation of the trace as outlined in Algorithm 19.*

9.2.2 A second strategy

We begin by observing that the partial traces in equation (27) can be rearranged in any order when viewed as operators on $\mathcal{K}$. This property is well-established and is briefly illustrated in Figure 4. In particular, we have

$$\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_0} = \mathrm{Tr}_{\alpha-1} \circ \cdots \circ \mathrm{Tr}_1 \circ \mathrm{Tr}_0.$$

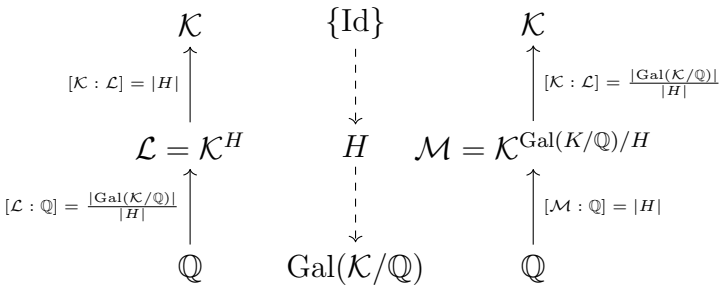

Figure 4: Field towers corresponding to the subgroups H and $\frac{\mathrm{Gal}(\mathcal{K}/\mathbb{Q})}{H}$ within $\mathrm{Gal}(\mathcal{K}/\mathbb{Q})$ are presented. The complete trace can be represented either as $\mathrm{Tr}_{\mathcal{L}/\mathbb{Q}} \circ \mathrm{Tr}_{\mathcal{K}/\mathcal{L}}$ or as $\mathrm{Tr}_{\mathcal{M}/\mathbb{Q}} \circ \mathrm{Tr}_{\mathcal{K}/\mathcal{M}}$. When all traces are extended to $\mathcal{K}$, it follows with a slight abuse of notations that $\mathrm{Tr}_{\mathcal{L}/\mathbb{Q}} \circ \mathrm{Tr}_{\mathcal{K}/\mathcal{L}} = \mathrm{Tr}_{\mathcal{K}/\mathcal{L}} \circ \mathrm{Tr}_{\mathcal{L}/\mathbb{Q}}$.

We will then undertake this packing process in two steps. In the first step, we will homomorphically construct a polynomial that incorporates μ_i as the coefficient at the $i\frac{M}{t^\beta}$-th position for all $0 \leq i \leq t^\beta \frac{N}{M} - 1$, while the other coefficients will be inconsequential (this step will require using the partial trace $\mathrm{Tr}_{\mathcal{K}_\beta/\mathbb{Q}}$ for each $c^{(i)}$). In the second step, we will apply the trace $\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_\beta}$ to this polynomial, thereby eliminating the remaining unwanted coefficients effectively.

To detail the first step, we first observe that the $i\frac{M}{t^\beta}$-th coefficient of the quantity

$$\mu(X) = \sum_{i=0}^{t^\beta \frac{N}{M}-1} \mu^{(i)}(X) X^{i\frac{M}{t^\beta}}$$

encrypted by

$$c(X) = \sum_{i=0}^{t^\beta \frac{N}{M}-1} c^{(i)}(X) X^{i\frac{M}{t^\beta}}$$

does not coincide with the message μ_i, although, by construction, the constant coefficient of $\mu^{(i)}$ is equal to μ_i for all $0 \leq i \leq t^\beta \frac{N}{M} - 1$. In order to construct a polynomial that has this desired property we first apply the trace operator

$$\mathrm{Tr}_{\mathcal{K}_\beta/\mathbb{Q}} = \mathrm{Tr}_{\beta-1} \circ \mathrm{Tr}_{\beta-2} \circ \cdots \circ \mathrm{Tr}_1 \circ \mathrm{Tr}_0$$

to each

$$\bar{\Omega}_0^*(X)\mu^{(i)}(X), \quad i = 0, \ldots, t^\beta \frac{N}{M} - 1,$$

in order to remove all its monomials whose exponents are non-zero multiples of $\frac{M}{t^\beta} = t^{\alpha-\beta}$ (refer to definitions (23) and (24) for these partial traces). Specifically:

- The operator Tr_0 removes all monomials with exponents that are non-zero multiples of $t^{\alpha-1}$.
- The operator Tr_1 eliminates all monomials whose exponents are multiples of $t^{\alpha-2}$ but not multiples of $t^{\alpha-1}$, while preserving those monomials with exponents that are multiples of $t^{\alpha-1}$.

- This elimination process continues in a similar fashion for the subsequent trace operators.
- Lastly, the operator $\mathrm{Tr}_{\beta-1}$ discards all monomials with exponents that are multiples of $t^{\alpha-\beta}$ but not multiples of $t^{\alpha-\beta+1}$, while keeping intact those monomials with exponents that are multiples of $t^{\alpha-\beta+1}$.

The resulting RLWEs are given by

$$\hat{c}^{(i)}(X) = \mathrm{RLWE}_{s(X)}\left(\mathrm{Tr}_{\mathcal{K}_\beta/\mathbb{Q}}\left(\bar{\Omega}_0^*(X)\mu^{(i)}(X)\right)\right).$$

These are then summed up to form

$$\hat{c}(X) = \sum_{i=0}^{t^\beta \frac{N}{M}-1} \hat{c}^{(i)}(X) X^{i\frac{M}{t^\beta}}.$$

Finally, the trace $\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_\beta}$ is applied, resulting in

$$\check{c}(X) = \mathrm{RLWE}_{s(X)}\left(\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_\beta}(\hat{c}(X))\right).$$

This process can be encapsulated in Algorithm 23. The total number of automorphisms

Algorithm 23 FastPack: packs $t^\beta \frac{N}{M}$ LWE-ciphertexts in an encrypted polynomial

Input: $c^{(i)}(X) = \mathrm{RLWE}_{s(X)}(\mu^{(i)}(X))$ with $(\mu^{(i)}(X))_0 = \mu_i$ and $1 \le \beta \le \alpha - 1$

1: **for** $i = 0$ to $t^\beta \frac{N}{M} - 1$ **do**
2: $\quad \hat{c}^{(i)}(X) \leftarrow \mathrm{PartialTrace}_{N/t^\beta \to 1}\left(\bar{\Omega}_0^*(X) c^{(i)}(X)\right)$
3: **end for**
4: $\hat{c}(X) \leftarrow \sum_{i=0}^{t^\beta \frac{N}{M}-1} X^{i\frac{M}{t^\beta}} \cdot \hat{c}^{(i)}(X)$
5: $\check{c}(X) \leftarrow \mathrm{PartialTrace}_{N \to N/t^\beta}\left(\hat{c}(X)\right)$
6: **Return** $\check{c}(X) = \mathrm{RLWE}_{s(X)}\left(\sum_{i=0}^{t^\beta \frac{N}{M}-1} \mu_i X^{i\frac{M}{t^\beta}}\right)$.

Note: The PartialTrace in steps 2 and 5 refers to the homomorphic computation of the trace as outlined in Algorithm 20.

required in this strategy is $(\beta-1)(t-1)t^\beta\frac{N}{M} + (\alpha-\beta)(t-1) + \sum_{j=1}^{r}(\ell_j - 1)$, which is less than the cost $t^\beta \frac{N}{M}\left((\alpha-1)(t-1) + \sum_{i=1}^{r}(\ell_j-1)\right)$ of the first approach (noting that $\beta \le \alpha - 1$).

9.2.3 The optimal strategy

We now present an additional refinement aimed at reducing the computational cost of the packing procedure. Our goal is to evaluate the following expression more efficiently:

$$\begin{aligned}\mu(X) &= \sum_{i=0}^{t^\beta \frac{N}{M}-1} \left(\mathrm{Tr}\left(\bar{\Omega}_0^*(X)\cdot\mu^{(i)}(X)\right)\right) X^{i\frac{M}{t^\beta}} \\ &= \sum_{i=0}^{t^\beta \frac{N}{M}-1} \mathrm{Tr}_{\mathcal{K}/\mathcal{K}_\beta} \circ \mathrm{Tr}_{\mathcal{K}_\beta/\mathbb{Q}}\left(\bar{\Omega}_0^*(X)\cdot\mu^{(i)}(X)\right) X^{i\frac{M}{t^\beta}}.\end{aligned}$$

We first utilize the fact that $\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_\beta}$ preserves the monomials with exponents that are multiples of $\frac{M}{t^\beta}$, which allows us to express:

$$\mu(X) = \mathrm{Tr}_{\mathcal{K}/\mathcal{K}_\beta}(\mu_\beta(X)), \quad \mu_\beta(X) = \sum_{i=0}^{t^\beta \frac{N}{M}-1} \mathrm{Tr}_{\mathcal{K}_\beta/\mathbb{Q}}\left(\bar{\Omega}_0^*(X)\cdot\mu^{(i)}(X)\right) X^{i\frac{M}{t^\beta}}.$$

For $0 \le i \le t^\beta \frac{N}{M} - 1$, we decompose (uniquely) the index i as follows:

$$i = i_0 + i_1 t + \ldots + i_{\beta-1}t^{\beta-1} = \sum_{k=0}^{\beta-1} i_k t^{\beta k}$$

where $0 \le i_k \le t-1$ for $k = 0, \ldots, \beta-2$ and $0 \le i_{\beta-1} \le t-2$. We can then rewrite the previous sum as:

$$\begin{aligned}\mu_\beta(X) &= \sum_{i_0,\ldots,i_{\beta-2}} \sum_{i_{\beta-1}} \left(\mathrm{Tr}_{\beta-1}\circ\cdots\circ\mathrm{Tr}_0\left(\bar{\Omega}_0^*(X)\cdot\mu^{(i)}(X)\right)\right) X^{i_0\frac{M}{t^\beta}+i_1\frac{M}{t^{\beta-1}}+\ldots+i_{\beta-1}\frac{M}{t}} \\ &= \sum_{i_0,\ldots,i_{\beta-2}} \mathrm{Tr}_{\beta-1}\circ\cdots\circ\mathrm{Tr}_1\left(\sum_{i_{\beta-1}}\mathrm{Tr}_0\left(\bar{\Omega}_0^*(X)\cdot\mu^{(i)}(X)\right)X^{i_{\beta-1}\frac{M}{t}}\right) X^{i_0\ldots+i_{\beta-2}\frac{M}{t^2}}\end{aligned}$$

Here, we have utilized the fact that $\mathrm{Tr}_{\beta-1}\circ\cdots\circ\mathrm{Tr}_1$ keeps intact those monomials with exponents that are multiples of $\frac{M}{t}$. Applying the same reasoning repetitively leads us to the final sum:

$$\sum_{i_0}\mathrm{Tr}_{\beta-1}\left(\ldots\left(\sum_{i_{\beta-2}}\mathrm{Tr}_1\left(\sum_{i_{\beta-1}}\mathrm{Tr}_0\left(\bar{\Omega}_0^*(X)\cdot\mu^{(i)}(X)\right)X^{i_{\beta-1}\frac{M}{t}}\right)X^{i_{\beta-2}\frac{M}{t^2}}\right)\ldots\right)X^{i_0\frac{M}{t^\beta}}$$

Let us denote

$$\mu_0^{(i)} = \bar{\Omega}_0^*(X)\mu^{(i)}(X), \quad i = 0, \ldots, t^\beta\frac{N}{M} - 1.$$

The various steps involved in the computation are as follows:

1. **Step** 1: Compute

$$\mu_1^{(i)}(X) = \sum_{i_{\beta-1}=0}^{t-2}\mathrm{Tr}_0\left(\mu_0^{(i+i_{\beta-1}t^{\beta-1})}(X)\right)X^{i_{\beta-1}\frac{M}{t}}, \quad i = 0, \ldots, t^{\beta-j}\frac{N}{M} - 1.$$

2. **Step** j (for $j = 2, \ldots, \beta-1$): Compute

$$\mu_j^{(i)}(X) = \sum_{i_{\beta-j}=0}^{t-1}\mathrm{Tr}_{j-1}\left(\mu_{j-1}^{(i+i_{\beta-j}t^{\beta-j})}(X)\right)X^{i_{\beta-j}\frac{M}{t^j}}, \quad i = 0, \ldots, t^{\beta-j}\frac{N}{M} - 1.$$

3. **Step** β: Compute $\mu_\beta(X) = \sum_{i_0=0}^{t-1}\mathrm{Tr}_{\beta-1}\left(\mu_{\beta-1}^{(i_0)}(X)\right)X^{i_0\frac{M}{t^\beta}}$.

4. **Final step**: Compute the final result $\mu(X) = \mathrm{Tr}_{\alpha-1}\circ\ldots\circ\mathrm{Tr}_\beta(\mu_\beta(X))$.

We summarize the individual steps in the algorithms below, first for cleartexts and then for ciphertexts. The associated computational cost can be readily assessed in terms of the occurrences of automorphisms (recall that $t-1=\prod_{i=1}^{r}\ell_r$ in this context) by reviewing the various steps outlined above:

$$\frac{N}{M}\left((t-1)t^{\beta-1}\left(\sum_{i=1}^{r}(\ell_j-1)\right)+\sum_{j=2}^{\beta-1}t(t-1)t^{\beta-j}\right)+t(t-1).$$

Algorithm 25 requires approximately $\beta\cdot t$-times fewer keyswitching operations than Algorithm 22 and is always strictly cheaper.

Algorithm 24 Packs $t^{\beta}\frac{N}{M}$ LWE-cleartexts in an encrypted polynomial

Input: $(\mu^{(i)}(X))$ with $(\mu^{(i)}(X))_0=\mu_i$, $i=0,\dots,t^{\beta}\frac{N}{M}-1$
Define $\mu_0^{(i)}=\bar{\Omega}_0^*(X)\mu^{(i)}(X),\quad i=0,\dots,t^{\beta}\frac{N}{M}-1$
for $i=0$ **to** $t^{\beta-1}\frac{N}{M}-1$ **do**
  Initialize: $\mu_1^{(i)}(X)\leftarrow 0$
  for $i_{\beta-1}=0$ **to** $t-2$ **do**
    $\mu_1^{(i)}(X)\leftarrow\mu_1^{(i)}(X)+\mathrm{Tr}_0\left(\mu_0^{(i+i_{\beta-1}t^{\beta-1})}(X)\right)X^{i_\beta\frac{M}{t}}$
  end for
end for
for $j=2$ **to** $\beta-1$ **do**
  for $i=0$ **to** $t^{\beta-j}\frac{N}{M}-1$ **do**
    Initialize: $\mu_j^{(i)}(X)\leftarrow 0$
    for $i_{\beta-j}=0$ **to** $t-1$ **do**
      $\mu_j^{(i)}(X)\leftarrow\mu_j^{(i)}(X)+\mathrm{Tr}_{j-1}\left(\mu_{j-1}^{(i+i_{\beta-j}t^{\beta-j})}(X)\right)X^{i_\beta\frac{M}{t^j}}$
    end for
  end for
end for
$\mu_\beta(X)\leftarrow 0$
for $i_0=0$ **to** $t-1$ **do**
  $\mu_\beta(X)\leftarrow\mu_\beta(X)+\mathrm{Tr}_{\beta-1}\left(\mu_{\beta-1}^{(i_0)}(X)\right)X^{i_\beta\frac{M}{t^\beta}}$
end for
Output: $\mu(X)\leftarrow\mathrm{Tr}_{\mathcal{K}/\mathcal{K}_\beta}(\mu_\beta(X))$

Appendix

Elementary operations in prime power cyclotomic rings

In this section of the paper, we explore the elementary operations necessary for manipulating the polynomials in $\mathcal{K}$ when the M-th cyclotomic polynomial Φ_M is defined by

$$\Phi_M(X)=\sum_{k=0}^{t-1}X^{k\frac{M}{t}} \tag{28}$$

so that

$$\Phi_M(X)(X^{\frac{M}{t}}-1)=X^M-1.$$

Algorithm 25 Packs $t^\beta \frac{N}{M}$ LWE-ciphertexts in an encrypted polynomial

Input: $\mathbf{c}^{(i)} = (\mathbf{a}^{(i)}, b^{(i)}) = \mathrm{LWE}_{\mathbf{s}}(\mu_i)$ for $0 \leq i < t^\beta \frac{N}{M}$
for $i = 0$ **to** $t^{\beta-1}\frac{N}{M} - 1$ **do**
  $a^{(0,i)}(X) \leftarrow \sum_{j=0}^{N-1} a_{j+1}^{(i)} \bar{\Omega}_j^*(X) \bmod \mathcal{R}, \quad b^{(0,i)}(X) \leftarrow \bar{\Omega}_0^*(X) b^{(i)}.$
end for
for $i = 0$ **to** $t^{\beta-1}\frac{N}{M} - 1$ **do**
  Initialize: $(a^{(1,i)}(X), b^{(1,i)}(X)) \leftarrow (0,0)$
  for $i_{\beta-1} = 0$ **to** $t-2$ **do**
    $(a^{(1,i)}(X), b^{(1,i)}(X)) \leftarrow (a^{(1,i)}(X), b^{(1,i)}(X)) + \mathrm{TR}_0\left(a^{(1,i)}(X), b^{(1,i)}(X)\right) X^{i_{\beta-1}\frac{M}{t}}$
  end for
end for
for $j = 2$ **to** $\beta - 1$ **do**
  for $i = 0$ **to** $t^{\beta-j}\frac{N}{M} - 1$ **do**
    Initialize: $(a^{(j,i)}(X), b^{(j,i)}(X)) \leftarrow (0,0)$
    for $i_{\beta-j} = 0$ **to** $t-1$ **do**
      $(a^{(j,i)}(X), b^{(j,i)}(X)) \leftarrow (a^{(j,i)}(X), b^{(j,i)}(X)) + \mathrm{TR}_{j-1}\left(a^{(j,i)}(X), b^{(j,i)}(X)\right) X^{i_{\beta-j}\frac{M}{t^j}}$
    end for
  end for
end for
Initialize: $(a^{(\beta,i)}(X), b^{(\beta,i)}(X)) \leftarrow (0,0)$
for $i_0 = 0$ **to** $t-1$ **do**
  $(a^{(\beta,i)}(X), b^{(\beta,i)}(X)) \leftarrow (a^{(\beta,i)}(X), b^{(\beta,i)}(X)) + \mathrm{Tr}_{\beta-1}\left(a^{(\beta,i)}(X), b^{(\beta,i)}(X)\right) X^{i_0\frac{M}{t^\beta}}$
end for
Output: $(a(X), b(X)) \leftarrow \mathrm{TR}_{\alpha-1} \circ \cdots \circ \mathrm{TR}_\beta(a^{(\beta,i)}(X), b^{(\beta,i)}(X))$

Taking the modulo

We begin by deriving a straightforward formula that allows to take the modulo Φ_M of any polynomial of degree less than or equal to $M-1$ within the ring $\mathcal{K}$. This can always be assumed as $X^M = 1 \bmod \Phi_M$.

Lemme 9.1 *Given $M = t^\alpha$ and $N = (t-1)t^{\alpha-1}$, consider the polynomial*

$$R(X) = \sum_{i=0}^{M-1} r_i X^i \in \mathcal{K}.$$

Then its unique representative modulo Φ_M of degree less or equal to $N-1$ is given by

$$(R \mod \Phi_M)(X) = \sum_{i=0}^{N-1} \left(r_i - r_{N+(i \bmod \frac{M}{t})} \right) X^i.$$

Proof. We first split R into two sums

$$R(X) = \sum_{k=0}^{M-1} r_k X^k = \sum_{k=0}^{N-1} r_k X^k + \sum_{k=N}^{M-1} r_k X^k = \sum_{k=0}^{N-1} r_k X^k + \sum_{k=0}^{M-N-1} r_{k+N} X^{k+N}.$$

and then use the following expression of Φ_M

$$X^N = -\sum_{j=0}^{t-2} X^{j\frac{M}{t}} \mod \Phi_M,$$

to rewrite R as

$$R(X) = \sum_{k=0}^{N-1} r_k X^k - \sum_{k=0}^{M-N-1} r_{k+N} \sum_{j=0}^{t-2} X^{k+j\frac{M}{t}} \mod \Phi_M.$$

Denoting

$$i = k + j\frac{M}{t},$$

and taking into account that $0 \le k \le M-N-1 = \frac{M}{t} - 1$, we have

$$0 \le i \le M-N-1+(t-2)\frac{M}{t} = N-1 \quad \text{and} \quad k = [i]_{\frac{M}{t}}.$$

Eventually,

$$R(X) = \sum_{i=0}^{N-1} \left(r_i - r_{N+i \bmod \frac{M}{t}} \right) X^i \mod \Phi_M.$$

Multiplication of polynomials

We now give the expression of the product modulo Φ_M of two polynomials.

Lemme 9.2 *Consider the following two polynomials of degrees less than $M-1$*

$$D(X) = \sum_{k=0}^{M-1} d_k X^k \in \mathcal{K} \quad \text{and} \quad P(X) = \sum_{k=0}^{M-1} a_k X^k \in \mathcal{K}.$$

The product $D \cdot P$ results in a polynomial in $\mathcal{K}$, and its unique representative modulo Φ_M of degree less than or equal to $N-1$ can be expressed as follows:

$$(D \cdot P)(X) = \sum_{i=0}^{N-1} \left(\sum_{j=0}^{M-1} d_j \left(a_{i-j} - a_{N-j+i \bmod \frac{M}{t}} \right) \right) X^i \quad \bmod \Phi_M,$$

where we adopt the convention that $a_{k+M} = a_k$ for all $k \in \mathbb{Z}$.

Proof. The product $D \cdot P$ modulo $X^M - 1$ can easily be written as

$$D(X)P(X) = \sum_{i=0}^{M-1} \left(\sum_{j=0}^{M-1} d_j a_{i-j} \right) X^i \mod X^M - 1,$$

with the convention that $a_{M+j} = a_j$ for all $j \in \mathbb{Z}$. Now, by applying Lemma 9.1 to $R = D \cdot P$, we obtain the result stated in this lemma.

Remark 9.3 *(for $t = 2$, $M = 2N$) If either of the polynomials D or P has a degree less than or equal to $(N-1)$ –let's assume D for instance– the summation over j can be restricted to indices between 0 and $N-1$. In the case where both polynomials have degrees less than or equal to $N-1$ and $M = 2N$, the sums can be further truncated. Specifically, we have:*

$$D(X)P(X) = \sum_{i=0}^{N-1} \left(\sum_{j=0}^{i} d_j a_{i-j} - \sum_{j=i+1}^{N-1} d_j a_{N+i-j} \right) X^i \mod X^N + 1.$$

Now, provided that the coefficients of D and P are extended by zeros for indices between N and $M-1$, and that both the negacyclicity and M-periodicity conventions $d_{N+i} = -d_i$, $a_{N+i} = -a_i$ and $d_{M+i} = d_i$, $a_{M+i} = a_i$ for all $i \in \mathbb{Z}$ hold, then we obtain the simple (and well-known) formula

$$D(X)P(X) = \sum_{i=0}^{N-1} \left(\sum_{j=0}^{N-1} d_j a_{i-j} \right) X^i \quad \bmod X^N + 1.$$

Estimates of the error growth resulting from a polynomial product

We now turn our attention to the number of non-zero terms in the coefficients of the product modulo Φ_M of two polynomials of degree less than $N-1$. Specifically, we seek to determine the number of non-zero terms in the sums given by

$$\sum_{j=0}^{N-1} d_j \left(a_{i-j} - a_{N-j+[i]_{\frac{M}{t}}} \right).$$

We will show that an accurate estimate of this quantity allows us to effectively evaluate how the noise is amplified during various encryption operations, including bootstrapping, key-switching, and packing. For instance, in the particular case where $t = 2$, this number equals N (as opposed to $2N$), which elucidates the multiplicative factor of N that appears in the noise levels of nearly all encryption operations involving polynomials. More specifically, we have the following lemma:

Lemme 9.4 *Let t be a prime integer, $M = t^\alpha$, with $\alpha \geq 2$, and $N = \varphi(M) = \frac{t-1}{t}M$. Consider the polynomials*

$$D = \sum_{i=0}^{N-1} d_i X^i, \qquad E = \sum_{i=0}^{N-1} e_i X^i$$

where the coefficients d_i and e_i are independent random variables with common standard deviations $\sigma(D)$ and $\sigma(E)$, respectively. Finally, let

$$P(X) = D(X) \cdot E(X) \ mod\ \Phi_M(X) = \sum_{i=0}^{N-1} p_i X^i$$

The variance of the random variable p_i can be expressed as follows:

$$\sigma(p_i)^2 = \left(N + i - [i]_{\frac{M}{t}} + \max\left(N - 1 - \frac{M}{t} - i, 0\right)\right)\sigma(D)^2\sigma(E)^2, \quad 0 \leq i \leq N-1.$$

Proof. For $0 \leq i \leq N-1$, let

$$A_i = \left\{0 \leq j \leq N-1, \quad s.\,t. \quad (i-j) \bmod M \leq N-1 \ and\ \left(N - j + [i]_{\frac{M}{t}}\right) \leq N-1\right\},$$

$$B_i = \left\{0 \leq j \leq N-1, \quad s.\,t. \quad (i-j) \bmod M \geq N \ and\ \left(N - j + [i]_{\frac{M}{t}}\right) \leq N-1\right\},$$

$$C_i = \left\{0 \leq j \leq N-1, \quad s.\,t. \quad (i-j) \bmod M \leq N-1 \quad and\ \left(N - j + [i]_{\frac{M}{t}}\right) \geq N\right\}.$$

We have from lemma 9.2 the following expression for the coefficient p_i:

$$p_i = \sum_{j \in A_i} d_j \left(e_{i-j} - e_{N-j+[i]_{\frac{M}{t}}}\right) - \sum_{j \in B_i} d_j e_{N-j+[i]_{\frac{M}{t}}} + \sum_{j \in C_i} d_j e_{i-j}.$$

This formulation relies on the facts

$$0 \leq N - j + [i]_{\frac{M}{t}} \leq M-1$$

and on the conventions $d_i = e_i = 0$ for $N \leq i \leq M-1$ and $d_{i+M} = d_i$, $e_{i+M} = e_i$ for all $i \in \mathbb{Z}$. We also note that

$$[i-j]_M \geq N \quad \Longrightarrow \quad N - j + [i]_{\frac{M}{t}} \leq N-1.$$

Given the mutual independence of the random variables d_i and e_i for $0 \leq i \leq N-1$, we can derive the expression for the variance of p_i:

$$\begin{aligned}\sigma(p_i)^2 = &\sum_{j \in A_i} \sigma(d_j)^2 \left(\sigma(e_{i-j})^2 + \sigma(e_{N-j+[i]_{\frac{M}{t}}})^2\right) \\ &+ \sum_{j \in B_i} \sigma(d_j)^2\sigma(e_{N-j+[i]_{\frac{M}{t}}})^2 + \sum_{j \in C_i} \sigma(d_j)^2\sigma(e_{i-j})^2.\end{aligned}$$

Note that $i - j \bmod M \neq N - j + i \bmod \frac{M}{t}$) for $0 \leq j \leq N-1$. We can easily determine the sets:

$$\begin{aligned}A_i &= \left\{j \ \text{ such that } \ 1 + [i]_{\frac{M}{t}} \leq j \leq i \ \text{ or } \ M - N + i + 1 \leq j \leq N-1\right\}, \\ B_i &= \left\{j \ \text{ such that } \ i+1 \leq j \leq \min(N-1, M-N+i)\right\}, \\ C_i &= \left\{j \ \text{ such that } \ 0 \leq j \leq [i]_{\frac{M}{t}}\right\}.\end{aligned}$$

From this, we derive the sizes:

$$|A_i| = i - ([i]_{\frac{M}{t}}) + \max(2N - M - i - 1, 0), |B_i| = \min(N-1, M-N+i) - i, |C_i| = 1 + ([i]_{\frac{M}{t}}).$$

Now, taking into account $\sigma(d_i) = \sigma(D)$ and $\sigma(e_i) = \sigma(E)$, we obtain:

$$\sigma(p_i)^2 = (2|A_i| + |B_i| + |C_i|)\,\sigma(D)^2\sigma(E)^2.$$

Finally noting that $|A_i| + |B_i| + |C_i| = N$, we conclude with the result of the lemma.

Remark 9.5 *We have*

$$\sigma((DP)_i)^2 = \sigma(D)^2\sigma(E)^2 \times \begin{cases} \frac{2t-3}{t-1}N - [i]_{\frac{M}{t}} - 1, & \text{if} \quad i \leq (t-2)\frac{M}{t} - 1, \\ \frac{2t-3}{t-1}N, & \text{if} \quad i \geq (t-2)\frac{M}{t}. \end{cases}$$

Interestingly, these estimates align precisely with Theorem 1 in [20]. In particular, we recover the formulas for variances that appear at the end of its proof, highlighting the consistency of our results within that framework.

Lemme 9.6 *Let $0 \leq i \leq N-1$ and $0 \leq \nu \leq M-1$. The following relation holds:*

$$\langle X^{-\nu}, \Omega_i^*(X)\rangle = \begin{cases} \delta_{M-\nu,i} & \text{if} \quad \frac{M}{t} + 1 \leq \nu \leq M-1 \\ -\delta_{\frac{M}{t}-\nu,[i]_{\frac{M}{t}}} & \text{if} \quad 1 \leq \nu \leq \frac{M}{t} \\ \delta_{i,0} & \text{if} \quad \nu = 0 \end{cases} \tag{29}$$

Proof. We have $-(M-1) \leq -\nu \leq 0$, so that $1 \leq M - \nu \leq M$. We thus have three cases: (i) $1 \leq M - \nu \leq N-1$: by definition of the dual basis, $\langle X^{-\nu}, \Omega_i^*(X)\rangle = \delta_{M-\nu,i}$; (ii) $N \leq M - \nu \leq M-1$: the application of the modulo gives $X^{M-\nu} = -\sum_{j=0}^{t-2} X^{M-\nu-j\frac{M}{t}}$ so that $\langle X^{-\nu}, \Omega_i^*(X)\rangle = -\sum_{j=0}^{t-2}\delta_{M-\nu-j\frac{M}{t},i} = -\delta_{\frac{M}{t}-\nu,[i]_{\frac{M}{t}}}$; (iii)$\nu = 0$: $\langle X^{-\nu}, \Omega_i^*(X)\rangle = \delta_{0,i}$

Corollary 9.7 *Let $0 \leq \nu \leq M-1$ and $e^*(X) = \sum_{i=0}^{N-1} e_i^*\Omega_i^*(X) \in \mathcal{K}^\vee$. The following relation holds:*

$$\langle X^{-\nu}, e^*(X)\rangle = \begin{cases} e^*_{M-\nu} & \text{if} \quad \frac{M}{t} + 1 \leq \nu \leq M-1 \\ -\sum_{j=1}^{t-1} e^*_{j\frac{M}{t}-\nu} & \text{if} \quad 1 \leq \nu \leq \frac{M}{t} \\ e^*_0 & \text{if} \quad \nu = 0 \end{cases} \tag{30}$$

Furthermore, if the coefficients of e^* are all centered random variables with the same standard deviation σ, then one has

$$\mathrm{Var}\left(\langle X^{-\nu}, e^*(X)\rangle\right) = \sigma^2\gamma(\nu) \tag{31}$$

with

$$\gamma(\nu) := \begin{cases} 1 & \text{if} \quad \frac{M}{t} + 1 \leq \nu \leq M-1 \text{ or } \nu = 0 \\ (t-1) & \text{if} \quad 1 \leq \nu \leq \frac{M}{t}. \end{cases} \tag{32}$$

Proof. The first part is a direct application of previous lemma. As for the second one, it is a simple consequence of the fact that all e_i^*'s are independent variables with the same variance σ^2.

Corollary 9.8 *Let* $P^*(X) = \sum_{i=0}^{N-1} p_i^* \Omega_i^*(X) \in \mathcal{K}^\vee$ *and* $Q(X) = \sum_{i=0}^{N-1} q_i \Omega_i(X) \in \mathcal{K}$. *Assume the coefficients of* P^* *and* Q *are all independent centered random variables with variances* $\mathrm{Var}(P^*)$ *and* $\mathrm{Var}(Q)$ *respectively. Then the following formula holds:*

$$\mathrm{Var}\left(\langle X^{-\nu}, P^*(X)Q(X)\rangle\right) = \Gamma(\nu)\,\mathrm{Var}(P^*)\,\mathrm{Var}(Q) \tag{33}$$

where

$$\Gamma(\nu) = \sum_{i=0}^{N-1} \gamma\left([\nu + i]_M\right).$$

References

[1] Martin R. Albrecht, Rachel Player, and Sam Scott, *On the concrete hardness of Learning with Errors*, Journal of Mathematical Cryptology, vol. 9, no. 3, pp. 169-203, October 2015. ISSN (Online): 1862-2984, ISSN (Print): 1862-2976.

[2] J.-C. Bajard, T. Gouget, B. Laigle, and M. Naya-Plasencia, *RNS Variant of FV-like Schemes*, in Proceedings of the 2017 IEEE International Symposium on Information Theory (ISIT), 2017, pp. 1596–1600. IEEE.

[3] Bernard, O., Joye, M., Smart, N.P., Walter, M. (2025). Drifting Towards Better Error Probabilities in Fully Homomorphic Encryption Schemes. In: Fehr, S., Fouque, PA. (eds) Advances in Cryptology - EUROCRYPT 2025. EUROCRYPT 2025. Lecture Notes in Computer Science, vol 15608. Springer, Cham.

[4] J-F. Biasse and L. Ruiz, *FHEW with efficient multibit bootstrapping*. In Progress in Cryptology - LATINCRYPT 2015 (Lecture Notes in Computer Science, Vol. 9230), K. Lauter and F. RodrÃguez-HenrÃquez (Eds.). Springer, 119-135. `https://doi.org/10.1007/978-3-319-22174-8_7`

[5] C. Bootland, W. Castryck, and F. Vercauteren, *On the Security of the Multivariate Ring Learning with Errors Problem*, in Proceedings of the 2020 ACM Conference on Computer and Communications Security (CCS), 2020, pp. 1237–1253. ACM.

[6] J.-P. Bossuat, A. Costache, C. Mouchet, L. Nürnberger, and J.R. Troncoso-Pastoriza, *Practical q-IND-CPA-D-Secure Approximate Homomorphic Encryption*, Cryptology ePrint Archive, Paper 2024/853, 2024.

[7] J. H. Cheon, A. Kim, and Y. Song, *A Simple, Efficient Bootstrapping Method for Fully Homomorphic Encryption over the Integers*, Cryptology ePrint Archive, 2021. `https://eprint.iacr.org/2021/180`

[8] P. Chartier, M. Koskas, M. Lemou, and F. Méhats, *Homomorphic Sign Evaluation with a RNS Representation of Integers*, *Advances in Cryptology-ASIACRYPT 2024*, Chung, K.M., Sasaki, Y. (eds.), Lecture Notes in Computer Science, vol. 15484, Springer, Singapore, 2025. `https://doi.org/10.1007/978-981-96-0875-1_9`.

[9] P. Chartier, M. Koskas, M. Lemou, and F. Méhats, *Fully Homomorphic Encryption on Large Integers*, Cryptology ePrint Archive, 2024.

[10] P. Chartier, M. Koskas, M. Lemou, and F. Méhats, *Method for Homomorphically Determining the Sign of a Message by Dilation, Associated Methods and Devices*, Patent no. WO2023242429 - 12/21/2023. Number and date of priority: FR2205957 - 06/17/2022.

[11] P. Chartier, M. Koskas, M. Lemou, and F. Méhats, *Homomorphic Encryption Method and Associated Devices and System*, Patent no. WO2022129979 - 06/23/2022. Number and date of priority: PCT/IB2020001147 - 12/18/2020.

[12] H. Chen, R. Gilad-Bachrach, K. Han, Z. Huang, A. Jalali, K. Laine, and K. Lauter, *Logistic Regression over Encrypted Data from Fully Homomorphic Encryption*, Cryptology ePrint Archive, Report 2018/462, 2018.

[13] H. Chen, W. Dai, M. Kim, and Y. Song, *Efficient Homomorphic Conversion Between (Ring) LWE Ciphertexts*, In: K. Sako and N. O. Tippenhauer (eds), Applied Cryptography and Network Security, ACNS 2021, Lecture Notes in Computer Science, vol. 12726, Springer, Cham.

[14] H. Chen, K. Lauter, and K. E. Stange, *Security Considerations for Galois Non-Dual RLWE Families*, Cryptology ePrint Archive, 2021. `https://eprint.iacr.org/2021/1620`

[15] I. Chillotti, N. Gama, M. Georgieva, and M. Izabachène, *Faster Fully Homomorphic Encryption: Bootstrapping in Less Than 0.1 Seconds*, Advances in Cryptology – ASIACRYPT 2016, pp. 3–33. Berlin, Heidelberg, Springer, 2016.

[16] I. Chillotti, N. Gama, M. Georgieva, and M. Izabachène, *TFHE: Fast Fully Homomorphic Encryption over the Torus*, Journal of Cryptology, 33(1), pp. 34–91, 2020.

[17] A. Costache, B. R. Curtis, E. Hales, S. Murphy, T. Ogilvie, and R. Player, *On the Precision Loss in Approximate Homomorphic Encryption*, Cryptology ePrint Archive, Report 2022/162, 2022.

[18] A. Costache and N. P. Smart, *Ring-LWE and its Applications to Lattice-Based Cryptography*, in Proceedings of the 2016 IEEE International Conference on Advanced Information Networking and Applications (AINA), 2016, pp. 163–170.

[19] E. Crockett and C. Peikert, *Challenges for Ring-LWE*, in Proceedings of the 2017 IEEE International Conference on the Theory and Application of Cryptography Technology (ACT), 2017, pp. 125–130. IEEE.

[20] De Micheli, G., Kim, D., Micciancio, D., Suhl, A. (2024). Faster Amortized FHEW Bootstrapping Using Ring Automorphisms. In: Tang, Q., Teague, V. (eds) Public-Key Cryptography - PKC 2024. PKC 2024. Lecture Notes in Computer Science, vol 14604. Springer, Cham. `https://doi.org/10.1007/978-3-031-57728-4_11`

[21] L. Ducas and D. Micciancio, *FHEW: Bootstrapping Homomorphic Encryption in Less Than a Second*, Advances in Cryptology – EUROCRYPT 2015, pp. 617–640. Berlin, Heidelberg, Springer, 2015.

[22] C. Gentry, *Fully Homomorphic Encryption Using Ideal Lattices*, 41st Annual ACM Symposium on Theory of Computing, pp. 169–178. ACM Press, 2009.

[23] R. Geelen and F. Vercauteren, *Fully Homomorphic Encryption for Cyclotomic Prime Moduli*, Cryptology ePrint Archive, Paper 2024/1587, 2024. `https://eprint.iacr.org/2024/1587`

[24] M. Joye, *SoK: Fully homomorphic encryption over the [discretized] torus PDF*, IACR Transactions on Cryptographic Hardware and Embedded Systems, 2022(4):661-692, 2022

[25] M. Joye, M. Walter, *Liberating TFHE: Programmable Bootstrapping with General Quotient Polynomials*, WAHC'22: Proceedings of the 10th Workshop on Encrypted Computing & Applied Homomorphic Cryptography Pages 1 - 11 https://doi.org/10.1145/3560827.3563376

[26] A. Kim, J. H. Cheon, and Y. Song, *A Simple, Efficient Bootstrapping Method for Fully Homomorphic Encryption over the Integers*, in Proceedings of the 2020 ACM Conference on Computer and Communications Security (CCS), 2020, pp. 123–145. ACM.

[27] E. Lee, J.-W. Lee, J.-S. No, and Y.-S. Kim, *Minimax Approximation of Sign Function by Composite Polynomial for Homomorphic Comparison*, IEEE Transactions on Dependable and Secure Computing, 2021.

[28] E. Lee, J.-W. Lee, Y.-S. Kim, and J.-S. No, *Optimization of Homomorphic Comparison Algorithm on RNS-CKKS Scheme*, IEEE Access, 10, pp. 26163-26176, 2022.

[29] Li, B., Micciancio, D.: On the security of homomorphic encryption on approximate numbers. In: Canteaut, A., Standaert, F.X. (eds.) EUROCRYPT 2021, Part I. LNCS, vol. 12696, pp. 648-677. Springer, Cham (Oct 2021).

[30] Vadim Lyubashevsky, Chris Peikert, and Oded Regev, *A Toolkit for Ring-LWE Cryptography*, Cryptology ePrint Archive, Paper 2013/293, 2013. https://eprint.iacr.org/2013/293

[31] Y. Lee, D. Micciancio, A. Kim, R. Choi, M. Deryabin, J. Eom, and D. Yoo, *Efficient FHEW Bootstrapping with Small Evaluation Keys, and Applications to Threshold Homomorphic Encryption*, Advances in Cryptology – EUROCRYPT 2023.

[32] D. Micciancio and Y. Polyakov, *Bootstrapping in FHEW-like Cryptosystems*, Association for Computing Machinery, New York, NY, USA, pp. 17–28, 2021.

[33] D. Micciancio and J. Sorrell, *Ring Packing and Amortized FHEW Bootstrapping*, Cryptology ePrint Archive, Paper 2018/532, 2018.

[34] O. Regev, "Learning with Errors," in *Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC)*, pp. 84–93, 2005, ACM. DOI: 10.1145/1060590.1060608.

[35] O. Regev, "On LWE and RLWE," in *Proceedings of the 2010 5th Theory of Cryptography Conference (TCC)*, vol. 5978, pp. 1–20, 2009.

[36] O. Regev, *On Lattices, Learning with Errors, Random Linear Codes, and Cryptography*, Journal of the ACM (JACM), 56(6), pp. 1–40, 2009.

[37] R. Schwerdt, L. Benz, W. Beskorovajnov, S. Eilebrecht, J. Müller-Quade, and A. Ottenhues, *Sender-binding Key Encapsulation*, Cryptology ePrint Archive, Paper 2023/127, 2023.

[38] M. Wang and F. Zhang, *On the Construction of Lattice-Based Fully Homomorphic Encryption Scheme with Prime Power Cyclotomic Polynomials*, International Journal of Information Security, 17(5), pp. 547–560, 2018. https://doi.org/10.1007/s10207-018-0437-1