

Arithmetic for Large-Characteristic Finite Fields in CKKS^{*}

Daehyun Jang and Junho Lee

Department of Mathematical Sciences, Seoul National University
jadh0309@snu.ac.kr, abab9579@snu.ac.kr

Abstract. Seuré and Suvanto (ePrint 2026/1102) recently showed that, for small-characteristic primes p , arithmetic over $\mathbb{F}_{p^r}$ can be homomorphically evaluated in CKKS via a technique they call *spectral encoding*. Their construction is, however, restricted to small characteristic: ciphertext multiplication amplifies the error by the operator norm of the multiplied plaintext. Under the spectral encoding, a field element in $\mathbb{F}_{p^r}$ is encoded to a plaintext with operator norm at most $O(rp)$. The error therefore grows by a factor of rp in the worst case, limiting the supported size of the characteristic prime p .

In this work, we propose *bi-spectral encoding*, which decomposes the coefficients of the plaintext polynomials used in the spectral encoding of Seuré and Suvanto. With a decomposition parameter d , a field element in $\mathbb{F}_{p^r}$ can be encoded to a plaintext with operator norm $O(rdp^{1/d})$, for a suitable choice of encoding parameters.

We give a comprehensive analysis of error growth and derive a bound on error amplification under multiplication. For fields with a 256-bit prime characteristic (secp256k1) and extension degrees 1, 2, and 4, we provide numerical simulations of error amplification and compute the maximum multiplication counts satisfying the decoding condition under stated error bounds.

We further present a bootstrapping procedure that reduces both the size of the plaintext and its noise.

Keywords: Fully Homomorphic Encryption · CKKS · Finite-Field Arithmetic · Large Precision Arithmetic

1 Introduction

Fully homomorphic encryption (FHE) allows computations on encrypted data without decrypting intermediate values. Different FHE schemes support different plaintext arithmetic: BGV [5] and BFV [4,11] provide exact arithmetic over finite rings, whereas CKKS [9] provides approximate arithmetic over vectors of real or complex numbers. Discrete variants of CKKS can also evaluate functions on discrete messages when the accumulated approximation error permits exact decoding. [10,2,1].

^{*} Working draft.

For finite-field arithmetic, Seuré and Suvanto [19] proposed *spectral encoding*, which represents elements of $\mathbb{F}_{p^r}$ as complex vectors while preserving addition and multiplication. Via this encoding, finite-field arithmetic can be performed within CKKS, as long as the error remains small enough to satisfy the decoding condition. Since CKKS bootstrapping can be faster than BFV or BGV bootstrapping for the corresponding arithmetic, this approach can offer performance advantages.

Arithmetic over $\mathbb{F}_{p^r}$ for large p is widely used in pairing-based cryptography [17]. Homomorphic evaluation of this arithmetic could support applications such as outsourcing signature generation while keeping the signing key encrypted. It could also be used for homomorphic proof generation, allowing zkSNARK proof generation to be delegated while keeping the witness private [12].

However, the spectral encoding is not well suited to large prime characteristic p . In the method of Seuré and Suvanto [19], the coefficients of a fresh plaintext can have magnitude on the order of p , while the size of the plaintext directly affects error amplification under multiplication. When two plaintexts containing error are multiplied, the error of each operand is multiplied by the other plaintext, whose size also grows as the computation proceeds. We therefore seek plaintexts for large-characteristic fields that limit error amplification.

1.1 Our Contributions

We apply the radix decomposition technique to the spectral encoding of Seuré and Suvanto [19]. Our contributions are as follows.

- We represent elements of $\mathbb{F}_{p^r}$ using bivariate polynomial ring in Y and B , where Y determines the field extension and B determines the radix decomposition of p . We give the encoding and decoding maps and prove the correctness of addition and multiplication under the coefficient-error condition.
- We characterize the error amplification by bounding the operator norm of the multiplication of fresh plaintexts in the bivariate polynomial ring. We tightly bound these factors in terms of the polynomials defining the bivariate polynomial ring. These bounds give a sufficient condition for decoding after successive multiplications by fresh plaintexts. For secp256k1, we compare the theoretical bounds with computed norms and determine the supported maximum number of multiplications under stated error assumptions.
- We propose a bootstrapping construction for refreshing the plaintext. The condition for further computation is stated in terms of bounds on the refreshed plaintext size and error.

1.2 Technical Overview

We first recall the encoding of Seuré and Suvanto [19]. For a message $m \in \mathbb{Z}[Y]/(p, F(Y))$, the plaintext has the form

$$\mu = \hat{m} + e, \quad \hat{m} = m + I \in \mathbb{Z}[Y]/(F(Y)), \quad I \in p(\mathbb{Z}[Y]/(F(Y))).$$

Table 1. Comparison of finite-field arithmetic. The extension-field column indicates support for $\mathbb{F}_{p^r}$ with $r > 1$. “Restricted” indicates that low-noise GBFV parameters are provided only for cyclotomic prime families such as Fermat and Goldilocks primes.

Work	Extension Field	Characteristic p	Ciphertext size
BGV/BFV [5,4,11]	○	Small	Small
MatriGear [7]	×	Restricted	Small
GBFV [14]	○	Restricted	Small
dBfV [18]	○	Large	Large
Seur�-Suvanto [19]	○	Small	Small
Wei et al. [21], Liu et al. [16]	×	Large	Small
Ours	○	Large	Small

We refer to m as a *message*, $\hat{m}$ as an *integer lift*, e as the (*coefficient*) *error*, and I as the *carry*. Multiplying two such plaintexts introduces the error term

$$\mu_0\mu_1 - (\mu_0 - e_0)(\mu_1 - e_1) = \mu_1e_0 + \mu_0e_1 - e_0e_1.$$

When the plaintexts have magnitude on the order of p , the characteristic p limits the number of multiplications before the coefficient error reaches $1/2$.

We apply the radix decomposition to the characteristic prime p . Choose a monic polynomial $G(B)$ of degree d with $G(b) = p$ for an integer b . We represent the integer plaintext $\hat{m}$ by

$$\hat{m}(Y, B) = \sum_{j=0}^{d-1} \hat{m}_j(Y) B^j, \quad \hat{m}(Y, b) \equiv m(Y) \pmod{p, F(Y)}.$$

We encode $\hat{m}$ to a complex vector by evaluation at the roots of F and G . This encoding preserves addition and multiplication. After decoding the result, rounding to the integer plaintext recovers the integer coefficients whenever the coefficient error is less than $1/2$. Then evaluation at $B = b$ and taking modulo $p, F(Y)$ recovers the field element.

For ease of notation, suppose that all input integer lifts are equal to m and all input errors are equal to e . Ignoring errors introduced by homomorphic operations, the first-order contribution of the input errors after multiplying $t + 1$ fresh plaintexts is bounded by

$$\|(t + 1)m^t e\|_\infty \leq (t + 1)\|m^t\|_{\text{op}}\|e\|_\infty.$$

In Theorem 1, we bound the operator norm of a product of k chosen lifts by a factor independent of k times Γ^k , with $\Gamma = O(db)$ for suitable F and G .

With our decomposition, $\Gamma = O(d \cdot p^{1/d})$ for suitable G . Without decomposition, $\Gamma = \Theta(p)$ for fixed F , making arithmetic with large p infeasible.

1.3 Related Work and Comparison

Discrete CKKS Discrete computations in CKKS require both an encoding of the messages and a way to control approximation errors. BLEACH [10] gives error-

cleaning procedures for discrete computations. Bae et al. [2] develop bootstrapping for binary data. Bae et al. and Alexandru et al. [3,1] evaluate functions given by discrete lookup tables during CKKS bootstrapping.

Finite Ring Arithmetic over CKKS Seuré and Suvanto [19] construct finite-field arithmetic using spectral encoding. Their implementation focuses on fields of small characteristic, such as $p = 2$ or up to $p = 31$.

For high-precision integer arithmetic, Kim [15] and Cha et al. [6] use digit representations and homomorphic carry procedures. Gao and Zheng [13] use triangle encoding to support arithmetic and Boolean operations on machine words. Shen et al. [20] extend triangle encoding to general digit bases and use overflow cancellation in conversion from arithmetic to Boolean representations.

BGV/BFV family Chen et al. [8] replace the integer plaintext modulus in BFV with a linear polynomial, enabling high-precision arithmetic with reduced noise growth. Cha et al. [7] and Geelen et al. [14] generalize BFV to polynomial plaintext moduli and give constructions for cyclotomic prime fields. The latter also supports field extensions. Peikert et al. [18] develop plaintext decomposition for high-precision exact arithmetic and analyze the resulting noise growth.

Concurrent Work As a concurrent work, both Wei et al. [21] and Liu et al. [16] extend Gao and Zheng’s work [13] into arbitrary integer moduli, including large prime p , and study the choice of defining polynomials, arithmetic, and conversion to digits with $O(1)$ bootstrapping. Our construction uses polynomial representations for extension fields and examines how the selected lifts affect repeated multiplication. This work was developed concurrently and independently with that of Wei et al. [21] and Liu et al. [16].

2 Preliminaries

2.1 Notation

For a prime p , $\mathbb{F}_{p^r}$ denotes the field with p^r elements. For monic polynomials, quotient-ring elements are identified with their unique representatives of degree below the modulus in each variable. The notation $\lfloor x \rfloor$ denotes nearest-integer rounding, applied coefficientwise to polynomials; ties may be resolved by a fixed rule. For a polynomial, $\|\cdot\|_1$ denotes the sum of the absolute values of its coefficients. We write I_r for the $r \times r$ identity matrix and $\otimes$ for the Kronecker product.

2.2 The CKKS Scheme

CKKS [9] supports approximate coordinatewise arithmetic on $\mathbb{C}^{N/2}$, where N is a power of two. Encoding maps a message to a ring element, and encryption produces a ciphertext of the ring element.

Encryption. CKKS is based on the RLWE assumption on the cyclotomic ring $\mathcal{R} = \mathbb{Z}[X]/(X^N + 1)$ and $\mathcal{R}_q = \mathcal{R}/q\mathcal{R}$. A ciphertext $(\beta, \alpha) \in \mathcal{R}_q^2$ encrypts $m \in \mathcal{R}$ under secret key $s \in \mathcal{R}$ if the RLWE equation $\beta + \alpha \cdot s = m + e \pmod{q}$ holds for small error $e \in \mathcal{R}$. We write $\text{sk} = s$ for the secret key.

Encoding. We use slot encoding and coefficient encoding. Both use a scaling factor Δ to reduce the relative effect of rounding errors. Let iDFT be the inverse of the discrete Fourier transform $\text{DFT}(m(X)) = (m(\zeta_{2N}^{5^j}))_{j \in [N/2]}$, where ζ_{2N} is a $2N$ -th root of unity.

- **Slot Encoding.** Given a message $\mathbf{z} \in \mathbb{C}^{N/2}$, the slot encoding with a scaling factor Δ outputs $\text{Ecd}_{\text{slot}}(\mathbf{z}) = \lfloor \Delta \cdot \text{iDFT}(\mathbf{z}) \rfloor \in \mathcal{R}$.
- **Coefficient Encoding.** Given a message $\mathbf{m} \in \mathbb{R}^N$, the coefficient encoding with scaling factor Δ outputs $\text{Ecd}_{\text{coeff}}(\mathbf{m}) = \sum_{i \in [N]} \lfloor \Delta \cdot m_i \rfloor X^i$.

We omit the scaling factor when the scale is clear from context.

Homomorphic Evaluation. CKKS supports the following homomorphic operations.

- **Add.** Given ciphertexts $\text{ct}, \text{ct}' \in \mathcal{R}_q^2$ encrypting $m, m' \in \mathcal{R}$, **Add** returns $\text{ct} + \text{ct}' \in \mathcal{R}_q^2$, which encrypts $m + m' \in \mathcal{R}$.
- **Mult.** Given ciphertexts $\text{ct}, \text{ct}' \in \mathcal{R}_q^2$ encrypting $m, m' \in \mathcal{R}$, **Mult** returns a ciphertext $\text{ct}'' \in \mathcal{R}_{q/\Delta}^2$ that encrypts $\lfloor mm'/\Delta \rfloor \in \mathcal{R}$. In slot encoding, **Mult** computes coordinatewise multiplication.
- **Auto.** Given a ciphertext $\text{ct} \in \mathcal{R}_q^2$ encrypting $m(X) \in \mathcal{R}$ and an odd index j , **Auto** returns a ciphertext $\text{ct}' \in \mathcal{R}_q^2$ encrypting $m(X^j)$. In slot encoding, **Auto** rotates the coordinates or takes their complex conjugates.

Mult consists of a tensor product, relinearization, and rescaling, and the errors from relinearization and rescaling are included in E_i in Section 4.1. Appendix B relates cyclotomic coefficient errors to errors in the plaintext algebra used below.

Bootstrapping. CKKS bootstrapping restores the ciphertext modulus after rescaling. A standard CKKS bootstrapping procedure consists of the following steps:

- **S2C** applies a homomorphic DFT to convert slot encoding to coefficient encoding. **S2C** receives a ciphertext encrypting $\text{Ecd}_{\text{slot}}(\mathbf{m})$, and returns a ciphertext that encrypts $m = \text{Ecd}_{\text{coeff}}(\mathbf{m})$.
- **ModRaise** lifts the ciphertext (β, α) from $\mathcal{R}_{q_0}$ to $\mathcal{R}_Q$ for $Q > q_0$, where it satisfies the identity $\beta + \alpha \cdot \text{sk} = m + q_0 \cdot I \pmod{Q}$ for some $I \in \mathcal{R}$.
- **C2S** is a homomorphic inverse DFT. **C2S** receives a ciphertext encrypting $m + q_0 \cdot I$ and returns a ciphertext encrypting $\text{Ecd}_{\text{slot}}(\mathbf{m} + q_0 \mathbf{I})$.
- **EvalMod** homomorphically removes the term $q_0 \mathbf{I}$ to obtain a ciphertext encrypting $\text{Ecd}_{\text{slot}}(\mathbf{m}) \in \mathcal{R}_q$ for some modulus $q_0 < q < Q$.

2.3 Mathematical Background

Below, let F, G be monic polynomials of degrees r, d , respectively, and let $\mathcal{P} := \mathbb{R}[Y, B]/(F(Y), G(B))$. We define two norms on $\mathcal{P}$.

Definition 1 (Coefficient norm). For $\mu(Y, B) \in \mathcal{P}$, we define its coefficient norm in the monomial basis $\{Y^i B^j\}_{0 \leq i < r, 0 \leq j < d}$ by

$$\|\mu\|_\infty := \left\| \sum_{i=0}^{r-1} \sum_{j=0}^{d-1} u_{ij} Y^i B^j \right\|_\infty = \max_{i,j} |u_{ij}|.$$

Definition 2 (Canonical norm). Suppose that F, G are squarefree over $\mathbb{R}$. We define the canonical norm of $\mu \in \mathcal{P}$ by

$$\|\mu\|_{\text{can}} := \max_{\substack{F(\xi)=0 \\ G(\omega)=0}} |\mu(\xi, \omega)|.$$

On real or complex coordinate vectors, $\|\cdot\|_\infty$ denotes the maximum absolute coordinate, using complex absolute values.

The following definition describes how a linear map between two normed spaces affects the norm.

Definition 3 (Operator norm). For a linear map T between real vector spaces equipped with the stated norms, define

$$\|T\|_{\infty \rightarrow \infty} := \sup_{x \neq 0} \frac{\|Tx\|_\infty}{\|x\|_\infty}.$$

In particular, for a real matrix acting between real coordinate spaces with the coordinate infinity norms,

$$\|T\|_{\infty \rightarrow \infty} = \max_i \sum_j |T_{ij}|.$$

For a fixed element $u \in \mathcal{P}$, multiplication by u is a linear map, so we can make the following definition.

Definition 4 (Multiplication operator norm). For $u \in \mathcal{P}$, define

$$\|u\|_{\text{op}} := \sup_{e \in \mathcal{P} \setminus \{0\}} \frac{\|ue\|_\infty}{\|e\|_\infty}.$$

The following definition will be used to bound the canonical norm of a lift.

Definition 5 (Lattices and successive minima). A full-rank lattice in an n -dimensional real vector space is the integer span of n linearly independent vectors. In fixed coordinates, a basis matrix A represents the lattice $A\mathbb{Z}^n$. For a full-rank lattice $\mathcal{L}$ and $1 \leq k \leq n$, $\lambda_k^{(\infty)}(\mathcal{L})$ is the least radius for which the $\|\cdot\|_\infty$ ball centered at zero contains k linearly independent vectors of $\mathcal{L}$.

3 Bi-Spectral Encoding for Large-Characteristic Finite Fields

3.1 Plaintext Decomposition

Let p be prime and let $F(Y) \in \mathbb{Z}[Y]$ be monic of degree r and irreducible modulo p . The message space is

$$\mathcal{M} = \mathbb{F}_{p^r} = \mathbb{Z}[Y]/(p, F(Y)).$$

In this one-variable representation, (centered) integer lifts $\hat{m}(Y) \in \mathbb{Z}[Y]/F(Y)$ have coefficients of absolute value at most $\lfloor p/2 \rfloor$. We use polynomials in Y, B to represent every message with a smaller coefficient infinity norm. Fix an integer $b \geq 2$. Using the radix decomposition of p , we represent a message by an integer polynomial whose evaluation at $B = b$ recovers the message. Choose a monic $G(B) \in \mathbb{Z}[B]$ of degree $d \geq 2$ with $G(b) = p$. Then define the integer plaintext ring

$$\mathcal{P}_{\mathbb{Z}} := \frac{\mathbb{Z}[Y, B]}{(F(Y), G(B))}.$$

Since $F(Y), G(B)$ are monic, every element of $\mathcal{P}_{\mathbb{Z}}$ has a unique polynomial representative

$$\sum_{i=0}^{r-1} \sum_{j=0}^{d-1} u_{ij} Y^i B^j, \quad u_{ij} \in \mathbb{Z}.$$

We use the same notation for an element and its polynomial representative, omitting $\text{mod}(F(Y), G(B))$.

An integer polynomial $\hat{m}(Y, B) \in \mathcal{P}_{\mathbb{Z}}$ is a lift of $m(Y) \in \mathcal{M}$ if $\hat{m}(Y, b) \equiv m(Y) \pmod{p, F(Y)}$. Since $G(b) = p$, reduction modulo $G(B)$ preserves evaluation at $B = b$ modulo $(p, F(Y))$. Evaluation at $B = b$ induces the surjective ring homomorphism

$$\pi_b : \mathcal{P}_{\mathbb{Z}} \longrightarrow \mathcal{M}, \quad \hat{m}(Y, B) \longmapsto \hat{m}(Y, b) \pmod{(p, F(Y))}.$$

Two lifts of the same field element differ by an element of $\ker \pi_b$. We define $\mathfrak{J}_b := \ker \pi_b$. Fix a lift $m(Y, B)$ of $m(Y)$. In expressions such as $m + I + e$, m also denotes this fixed integer polynomial representative of the message. The integer plaintexts representing $m(Y)$ form the coset

$$\pi_b^{-1}(m(Y)) = m(Y, B) + \mathfrak{J}_b.$$

For an integer plaintext $\hat{m}(Y, B) \in \pi_b^{-1}(m(Y))$, we can write

$$\hat{m}(Y, B) = m(Y, B) + I(Y, B), \quad I(Y, B) \in \mathfrak{J}_b.$$

To bound the coefficient norm of a lift for each message, we first identify $\mathfrak{J}_b$ and give a lattice basis.

Proposition 1. $\mathfrak{J}_b = \ker \pi_b = (B - b)\mathcal{P}_{\mathbb{Z}}$.

Proof. The quotient rings satisfy

$$\frac{\mathbb{Z}[Y, B]/(F(Y), G(B))}{(B - b)} \cong \frac{\mathbb{Z}[Y, B]}{(F(Y), G(B), B - b)} \cong \frac{\mathbb{Z}[Y]}{(F(Y), G(b))} = \frac{\mathbb{Z}[Y]}{(F(Y), p)}.$$

The quotient map is evaluation at $B = b$ modulo $p, F(Y)$, namely π_b . Hence $\ker \pi_b = (B - b)\mathcal{P}_{\mathbb{Z}}$. $\square$

The residue classes of $Y^i B^j$, $0 \leq i < r$ and $0 \leq j < d$, form a $\mathbb{Z}$ -basis of $\mathcal{P}_{\mathbb{Z}}$. Coefficients in this basis identify the additive group $\mathcal{P}_{\mathbb{Z}}$ with $\mathbb{Z}^{rd}$ and $\mathfrak{J}_b$ with a full-rank lattice $\mathcal{L}_{\text{coeff}}$ in $\mathbb{R}^{rd}$.

Proposition 2 (Ideal Lattice). Write $G(B) = B^d + \sum_{k=0}^{d-1} g_k B^k$. In the ordered coefficient basis $(Y^i B^j)_{\substack{0 \leq i < r, \\ 0 \leq j < d}}$, with j varying first, $\mathcal{L}_{\text{coeff}} = (I_r \otimes M_G)\mathbb{Z}^{rd}$, where

$$M_G = \begin{pmatrix} -b & 0 & \cdots & 0 & -g_0 \\ 1 & -b & \ddots & \vdots & -g_1 \\ 0 & \ddots & \ddots & 0 & \vdots \\ \vdots & \ddots & 1 & -b & -g_{d-2} \\ 0 & \cdots & 0 & 1 & -(g_{d-1} + b) \end{pmatrix}, \quad I_r \otimes M_G = \begin{pmatrix} M_G & 0 & \cdots & 0 \\ 0 & M_G & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & M_G \end{pmatrix}$$

and I_r is the $r \times r$ identity matrix. Consequently, $\det(\mathcal{L}_{\text{coeff}}) = p^r$.

Proof. By Proposition 1, $\mathfrak{J}_b$ is the image of multiplication by $B - b$ on $\mathcal{P}_{\mathbb{Z}}$. For $0 \leq j < d - 1$,

$$(B - b)B^j = B^{j+1} - bB^j,$$

whereas reduction modulo $G(B)$ gives

$$(B - b)B^{d-1} = -\sum_{k=0}^{d-2} g_k B^k - (g_{d-1} + b)B^{d-1}.$$

The coefficient vectors are the columns of M_G . Multiplication by $B - b$ leaves the exponent of Y unchanged, so M_G acts on each block corresponding to Y^i . Hence the multiplication matrix is $I_r \otimes M_G$.

The matrix of multiplication by B has characteristic polynomial G , so $|\det M_G| = |G(b)| = p$. The columns of $I_r \otimes M_G$ are therefore linearly independent and form a basis of the image lattice $\mathfrak{J}_b$. Thus

$$\det(\mathcal{L}_{\text{coeff}}) = |\det(I_r \otimes M_G)| = p^r.$$

$\square$

Rounding coordinates in the basis $I_r \otimes M_G$ gives a bound on the coefficient norm of a lift for every message.

Proposition 3. *Every $m(Y) \in \mathcal{M}$ has an integer lift $\widehat{m}(Y, B) \in \mathcal{P}_{\mathbb{Z}}$ satisfying*

$$\|\widehat{m}\|_{\infty} \leq \frac{1}{2} \|M_G\|_{\infty \rightarrow \infty} \leq \frac{1 + |b| + \max_{0 \leq k < d} |g_k|}{2}.$$

Proof. Let $x \in \mathbb{Z}^{rd}$ be the coefficient vector of any lift of $m(Y)$. Since $\det(I_r \otimes M_G) \neq 0$, choose $z \in \mathbb{Z}^{rd}$ by rounding each coordinate of $(I_r \otimes M_G)^{-1}x$ to a nearest integer. The integer vector $x - (I_r \otimes M_G)z$ represents a lift of the same message because $(I_r \otimes M_G)z \in \mathfrak{J}_b$. Moreover,

$$\|x - (I_r \otimes M_G)z\|_{\infty} \leq \|I_r \otimes M_G\|_{\infty \rightarrow \infty} \|(I_r \otimes M_G)^{-1}x - z\|_{\infty} \leq \frac{1}{2} \|M_G\|_{\infty \rightarrow \infty}.$$

Each absolute row sum of M_G is at most $1 + |b| + \max_{0 \leq k < d} |g_k|$, which proves the second bound. $\square$

Consequently, if the chosen b, G satisfy

$$|b| + \max_{0 \leq k < d} |g_k| = O(p^{1/d}),$$

every field element can be encoded using an integer lift of coefficient norm $O(p^{1/d})$, compared with the worst-case $\lfloor p/2 \rfloor$ in the one-variable representation. Section 4 bounds the multiplication operator norms of plaintext powers to analyze the resulting error amplification.

3.2 Encoding and Decoding in CKKS

Recall the real plaintext algebra $\mathcal{P} = \mathbb{R}[Y, B]/(F(Y), G(B))$, with $\mathcal{P}_{\mathbb{Z}} \subset \mathcal{P}$. If F, G are squarefree over $\mathbb{R}$, then root evaluation gives an $\mathbb{R}$ -algebra isomorphism

$$V^{-1} : \mathcal{P} \longrightarrow \mathbb{C}^a \times \mathbb{R}^c, \quad \mu(Y, B) \longmapsto (\mu(\xi_i, \omega_j))_{i,j},$$

where ξ_i and ω_j are roots of F and G , respectively. The evaluation coordinates contain one pair from each orbit under $(\xi, \omega) \mapsto (\bar{\xi}, \bar{\omega})$. There are a nonreal conjugate pairs and c real root pairs, so $2a + c = rd$. If F and G factor over $\mathbb{R}$ into irreducible quadratic factors only, then $c = 0$ and $2a = rd$. Let V denote the inverse of V^{-1} . For $\mu \in \mathcal{P}$, root evaluation gives $\|\mu\|_{\text{can}} = \|V^{-1}\mu\|_{\infty}$. Denote encoding and decoding by Ecd and Dcd , respectively. Let N be a power of two and let $\mathcal{R} = \mathbb{Z}[X]/(X^N + 1)$ be the CKKS cyclotomic ring. Assuming $a + c \leq N/2$, the encoding map is

$$\text{Ecd} : \mathbb{F}_{p^r} \cong \mathcal{P}_{\mathbb{Z}}/\mathfrak{J}_b \xrightarrow{\text{lift}} \mathcal{P}_{\mathbb{Z}} \hookrightarrow \mathcal{P} \xrightarrow{V^{-1}} \mathbb{C}^a \times \mathbb{R}^c \hookrightarrow \mathbb{C}^{N/2} \xrightarrow{[\Delta \cdot \text{iDFT}]} \mathcal{R}.$$

The encoded polynomial is then encrypted using RLWE encryption. After decryption, divide by Δ , apply DFT, and select the coordinates used in encoding. Take the real parts of the coordinates corresponding to real root pairs, then apply V . The recovered plaintext $\mu(Y, B) \in \mathcal{P}$ has the form

$$\mu(Y, B) = \widehat{m}(Y, B) + e(Y, B), \quad \widehat{m}(Y, B) = m(Y, B) + I(Y, B).$$

Here $e \in \mathcal{P}$ includes encoding, encryption and computation errors.

Correct decoding depends on rounding from $\mathcal{P}$ to $\mathcal{P}_{\mathbb{Z}}$. Each coefficient of the unique polynomial representative of μ is rounded to the nearest integer. If $\|e\|_{\infty} < 1/2$, then $\lfloor \mu \rfloor = \hat{m}$ and thus $\pi_b(\lfloor \mu \rfloor) = \pi_b(\hat{m}) = m \in \mathcal{M} = \mathbb{F}_{p^r}$. Thus coefficient error strictly less than $1/2$ guarantees correctness.

3.3 Homomorphic Operations

Let the input plaintexts be $\mu = \hat{m} + e, \mu' = \hat{m}' + e'$, where $\hat{m} = m + I$ and $\hat{m}' = m' + I'$ belong to $\mathcal{P}_{\mathbb{Z}}$.

Proposition 4 (Addition). *For ciphertexts at the same modulus and scale, direct ciphertext addition has plaintext*

$$\mu + \mu' = (\hat{m} + \hat{m}') + (e + e').$$

If $\|e + e'\|_{\infty} < 1/2$, decoding yields $\pi_b(\hat{m}) + \pi_b(\hat{m}')$.

Proposition 5 (Multiplication). *Suppose that ciphertext multiplication, including relinearization and rescaling, introduces an additional error $E_{\text{mult}} \in \mathcal{P}$. The output plaintext is*

$$\mu_{\text{out}} = \hat{m}\hat{m}' + e_{\text{out}}, \quad e_{\text{out}} = \hat{m}e' + e\hat{m}' + ee' + E_{\text{mult}}.$$

If $\|e_{\text{out}}\|_{\infty} < 1/2$, decoding yields $\pi_b(\hat{m})\pi_b(\hat{m}')$.

Applying V converts the additional CKKS multiplication error $\varepsilon_{\text{mult}} \in \mathbb{C}^{N/2}$ to the plaintext error $E_{\text{mult}} \in \mathcal{P}$, with

$$\|E_{\text{mult}}\|_{\infty} \leq \|V\|_{\infty \rightarrow \infty} \|\varepsilon_{\text{mult}}\|_{\infty}.$$

The product of $\|V\|_{\infty \rightarrow \infty}$ and $\|\varepsilon_{\text{mult}}\|_{\infty}$ bounds the contribution of the additional error to the decoding condition.

3.4 One-Variable Cases

For $F(Y) = Y - p$, the message field and plaintext algebra satisfy

$$\frac{\mathbb{Z}[Y]}{(p, Y - p)} \cong \mathbb{F}_p, \quad \frac{\mathbb{R}[Y, B]}{(Y - p, G(B))} \cong \frac{\mathbb{R}[B]}{(G(B))}.$$

The degree-bounded representatives are independent of Y . The choice $F(Y) = Y - p$ gives the one-variable polynomial representation for a prime modulus, as in the arithmetic representation of Liu et al. [16] and Wei et al. [21]. More generally, choosing $G(b) = M$ for any integer $M > 1$ gives $\mathcal{P}_{\mathbb{Z}}/(B - b)\mathcal{P}_{\mathbb{Z}} \cong \mathbb{Z}/M\mathbb{Z}$, recovering their polynomial representation for arbitrary integer moduli.

Alternatively, allowing $G(B) = B$ and $b = p$ gives $G(b) = p$ and

$$\mathcal{P} \cong \frac{\mathbb{R}[Y]}{(F(Y))}, \quad \mathfrak{I}_p = p(\mathbb{Z}[Y]/(F(Y))).$$

Multiplication by $B - p$ is multiplication by $-p$, so $M_G = (-p)$ is a 1×1 matrix. Evaluation at the only root $B = 0$ leaves the Y -root evaluation only. Therefore, the choice $G(B) = B$ recovers the plaintext algebra and decoding map of Seuré and Suvanto [19].

Taking both $F(Y) = Y - p$ and $G(B) = B$, with $b = p$, gives $\mathcal{P}_{\mathbb{Z}} \cong \mathbb{Z}$ and $\mathcal{P} \cong \mathbb{R}$. The encoding therefore reduces to

$$\mathbb{Z}/p\mathbb{Z} \xrightarrow{\text{lift}} \mathbb{Z} \hookrightarrow \mathbb{R} \hookrightarrow \mathbb{C}.$$

For $p = 2$, choosing the representatives $0, 1$ encodes bits as real numbers in CKKS slots, as in the binary computations of BLEACH [10].

4 Error Analysis

We analyze error propagation under successive multiplications by fresh plaintexts. Multiplying $t + 1$ inputs requires t multiplications. The coefficient norm of the error determines decoding correctness, while multiplication operator norms bound its amplification. We bound the operator norms of plaintext products through the canonical norm, then choose lifts with small canonical norm.

4.1 Error Propagation and Multiplication Operators

Let the fresh plaintexts be

$$\mu_j = m_j + I_j + e_j = \hat{m}_j + e_j, \quad 0 \leq j \leq t,$$

where $\hat{m}_j \in \mathcal{P}_{\mathbb{Z}}$ and $e_j \in \mathcal{P}$. Their sequential product satisfies

$$\nu_0 = \mu_0, \quad \nu_i = \nu_{i-1}\mu_i + E_i \quad (1 \leq i \leq t),$$

where $E_i \in \mathcal{P}$ is the error introduced by the i th multiplication via relinearization and rescaling. Write $P_i = \prod_{j=0}^i \hat{m}_j$ for the exact integer product. All empty products below equal 1.

Proposition 6 (Error under successive multiplication). *For $t \geq 1$, the total error satisfies*

$$\nu_t - P_t = \sum_{j=0}^t \left(\prod_{\ell=0}^{j-1} \hat{m}_\ell \right) \left(\prod_{\ell=j+1}^t \mu_\ell \right) e_j + \sum_{i=1}^t \left(\prod_{\ell=i+1}^t \mu_\ell \right) E_i. \quad (1)$$

Consequently,

$$\|\nu_t - P_t\|_\infty \leq \sum_{j=0}^t \left\| \left(\prod_{\ell=0}^{j-1} \hat{m}_\ell \right) \left(\prod_{\ell=j+1}^t \mu_\ell \right) \right\|_{\text{op}} \|e_j\|_\infty + \sum_{i=1}^t \left\| \prod_{\ell=i+1}^t \mu_\ell \right\|_{\text{op}} \|E_i\|_\infty. \quad (2)$$

If the right-hand side is less than $1/2$, decoding ν_t recovers $\prod_{j=0}^t m_j$.

Proof. The recurrence gives

$$\nu_t = \prod_{j=0}^t \mu_j + \sum_{i=1}^t \left(\prod_{\ell=i+1}^t \mu_\ell \right) E_i.$$

Replacing the factors μ_j by $\hat{m}_j$ one at a time gives

$$\prod_{j=0}^t \mu_j - \prod_{j=0}^t \hat{m}_j = \sum_{j=0}^t \left(\prod_{\ell=0}^{j-1} \hat{m}_\ell \right) (\mu_j - \hat{m}_j) \left(\prod_{\ell=j+1}^t \mu_\ell \right).$$

Substituting $\mu_j - \hat{m}_j = e_j$ gives (1). The triangle inequality and the multiplication operator norm give (2). $\square$

4.2 Bounding Products through the Canonical Norm

Ignoring the operation errors, the term linear in the input errors in Proposition 6 is

$$\sum_{j=0}^t \left(\prod_{\substack{0 \leq \ell \leq t \\ \ell \neq j}} \hat{m}_\ell \right) e_j.$$

We therefore bound $\|\prod \hat{m}_\ell\|_{\text{op}}$.

Root evaluation bounds multiplication operator norms through the canonical norm $\|a\|_{\text{can}} = \|V^{-1}a\|_{\infty}$.

Proposition 7. *Suppose that F and G are squarefree. For every integer $k \geq 1$ and $a_1, \dots, a_k \in \mathcal{P}$,*

$$\left\| \prod_{j=1}^k a_j \right\|_{\text{can}} \leq \left\| \prod_{j=1}^k a_j \right\|_{\text{op}} \leq \|V\|_{\infty \rightarrow \infty} \|V^{-1}\|_{\infty \rightarrow \infty} \prod_{j=1}^k \|a_j\|_{\text{can}}.$$

Proof. Let $a = \prod_{j=1}^k a_j$. Evaluation preserves multiplication. Thus, for every $e \in \mathcal{P}$,

$$V^{-1}(ae) = (a(\xi, \omega)e(\xi, \omega))_{(\xi, \omega)},$$

where the coordinates run over the selected root pairs. Hence

$$\|ae\|_{\infty} \leq \|V\|_{\infty \rightarrow \infty} \|a\|_{\text{can}} \|V^{-1}\|_{\infty \rightarrow \infty} \|e\|_{\infty}.$$

Taking the supremum over $e \neq 0$ and using $\|a\|_{\text{can}} \leq \prod_{j=1}^k \|a_j\|_{\text{can}}$ proves the upper bound. For each root pair, evaluation is a nonzero linear functional satisfying $(ae)(\xi, \omega) = a(\xi, \omega)e(\xi, \omega)$. Therefore $a(\xi, \omega)$ is an eigenvalue of the transpose of the multiplication matrix of a . The absolute value of each eigenvalue is at most the matrix operator norm. Taking the maximum over the root pairs gives the lower bound. $\square$

For any $a \in \mathcal{P}$, Proposition 7 determines the exponential growth rate of $\|a^k\|_{\text{op}}$. For $a \neq 0$, after taking k -th roots, the ratio between the upper and lower bounds is

$$\left(\|V\|_{\infty \rightarrow \infty} \|V^{-1}\|_{\infty \rightarrow \infty}\right)^{1/k},$$

which tends to 1 as k increases.

Proposition 7 uses the operator norms of evaluation and interpolation and the canonical norms of the factors. We first bound the operator norms of evaluation and interpolation, which are independent of k .

Lemma 1. *The root-evaluation map satisfies*

$$\|V^{-1}\|_{\infty \rightarrow \infty} \leq \max_{\substack{F(\xi)=0 \\ G(\omega)=0}} \left(\sum_{i=0}^{r-1} |\xi|^i \right) \left(\sum_{j=0}^{d-1} |\omega|^j \right).$$

Proof. For $a(Y, B) = \sum_{i,j} a_{ij} Y^i B^j \in \mathcal{P}$, the coefficient norm gives

$$|a(\xi, \omega)| \leq \sum_{i,j} |a_{ij}| |\xi|^i |\omega|^j \leq \|a\|_{\infty} \left(\sum_{i=0}^{r-1} |\xi|^i \right) \left(\sum_{j=0}^{d-1} |\omega|^j \right).$$

Take the maximum over the selected root pairs and apply the definition of the operator norm in Section 2. $\square$

The bound in Lemma 1 depends on the magnitudes of the roots of F, G . Cauchy's bound estimates the root magnitudes from the coefficients.¹

Lemma 2 (Cauchy bound). *Let*

$$F(Y) = Y^r + \sum_{j=0}^h f_j Y^j, \quad 0 \leq h < r, \quad S = \sum_{j=0}^h |f_j| > 0.$$

Every root ξ of F satisfies $|\xi| \leq \max\{1, S^{1/(r-h)}\}$.

Proof. If $|\xi| \leq 1$, the bound holds. Otherwise, $F(\xi) = 0$ gives

$$|\xi|^r = \left| \sum_{j=0}^h f_j \xi^j \right| \leq \sum_{j=0}^h |f_j| |\xi|^j \leq S |\xi|^h.$$

Dividing by $|\xi|^h$ proves the bound. $\square$

¹ Seuré and Suvanto [19] derive an error bound when the degree gap $r - h$ is sufficiently large. Cauchy's bound relates the degree gap to root magnitudes, which bound the operator norm of root evaluation by Lemma 1.

If F has at most s nonzero coefficients below degree r , each of absolute value at most A , Lemma 2 gives $|\xi| \leq \max\{1, (sA)^{1/(r-h)}\}$. For a monic polynomial F of degree r with $F(Y) \neq Y^r$, define its *degree gap* by

$$\Delta_F = r - \deg(F - Y^r).$$

For $F(Y) = Y^r$, set $\Delta_F := r$. Define Δ_G analogously, with $\Delta_G := d$ when $G(B) = B^d$.

Combining Lemmas 1 and 2 gives the following coefficient bound.

Corollary 1. *For $F(Y) = Y^r + \sum_{i=0}^{h_F} f_i Y^i$, $G(B) = B^d + \sum_{j=0}^{h_G} g_j B^j$, with $0 \leq h_F < r$ and $0 \leq h_G < d$, we have*

$$\|V^{-1}\|_{\infty \rightarrow \infty} \leq \left(\sum_{i=0}^{r-1} \|F\|_1^{i/(\Delta_F)} \right) \left(\sum_{j=0}^{d-1} \|G\|_1^{j/(\Delta_G)} \right).$$

Proof. Since the coefficient 1-norm includes the leading coefficient, Lemma 2 gives

$$|\xi| \leq \|F\|_1^{1/(\Delta_F)}, \quad |\omega| \leq \|G\|_1^{1/(\Delta_G)}.$$

If either polynomial consists only of its leading monomial, the corresponding root is zero and the same inequality holds. Substitute these inequalities into Lemma 1. $\square$

The norm $\|V\|_{\infty \rightarrow \infty}$ bounds error amplification when recovering coefficients from root evaluations. We bound $\|V\|_{\infty \rightarrow \infty}$ by expressing the coefficients through Lagrange interpolation.

Proposition 8. *Suppose that F, G are monic and squarefree. Write*

$$\frac{F(Y)}{(Y - \xi)F'(\xi)} = \sum_{i=0}^{r-1} \ell_{\xi,i} Y^i, \quad \frac{G(B)}{(B - \omega)G'(\omega)} = \sum_{j=0}^{d-1} k_{\omega,j} B^j.$$

Then

$$\|V\|_{\infty \rightarrow \infty} \leq \left(\max_{0 \leq i < r} \sum_{F(\xi)=0} |\ell_{\xi,i}| \right) \left(\max_{0 \leq j < d} \sum_{G(\omega)=0} |k_{\omega,j}| \right). \quad (3)$$

The sums range over all roots of F and G , respectively.

Proof. For $a(Y, B) = \sum_{i,j} a_{ij} Y^i B^j \in \mathcal{P}$, Lagrange interpolation gives

$$a_{ij} = \sum_{\substack{F(\xi)=0 \\ G(\omega)=0}} a(\xi, \omega) \ell_{\xi,i} k_{\omega,j}.$$

The sum includes both members of each conjugate pair. Restoring the omitted evaluation coordinates by complex conjugation preserves the canonical norm. Hence

$$|a_{ij}| \leq \|a\|_{\text{can}} \left(\sum_{F(\xi)=0} |\ell_{\xi,i}| \right) \left(\sum_{G(\omega)=0} |k_{\omega,j}| \right).$$

Taking the maximum over i, j and then the supremum of $\|a\|_\infty / \|a\|_{\text{can}}$ over nonzero a proves the bound. $\square$

The interpolation coefficients depend on the distances between roots through the denominators

$$|F'(\xi)| = \prod_{\substack{F(\xi')=0 \\ \xi' \neq \xi}} |\xi - \xi'|, \quad |G'(\omega)| = \prod_{\substack{G(\omega')=0 \\ \omega' \neq \omega}} |\omega - \omega'|.$$

For any plaintexts $a_1, \dots, a_k \in \mathcal{P}$ with $\|a_j\|_{\text{can}} \leq A$, Proposition 7 gives

$$\left\| \prod_{j=1}^k a_j \right\|_{\text{op}} \leq \|V\|_{\infty \rightarrow \infty} \|V^{-1}\|_{\infty \rightarrow \infty} A^k.$$

The bound applies to chosen lifts and to plaintexts obtained after bootstrapping (Section 5). A larger value of A increases the error bound and may reduce the multiplication count permitted by the decoding condition.

4.3 Choosing Lifts with Small Canonical Norm

The bounds on $\|V\|_{\infty \rightarrow \infty}$ and $\|V^{-1}\|_{\infty \rightarrow \infty}$ depend only on F, G . It remains to bound the canonical norm of an available lift for each message. We use the root-evaluation image of the coefficient lattice from Section 3.1:

$$\mathcal{L} := V^{-1} \mathcal{L}_{\text{coeff}} = V^{-1} (I_r \otimes M_G) \mathbb{Z}^{rd}.$$

Both lattices represent the same carry ideal $\mathfrak{J}_b$ in different coordinates. Adding an element of $\mathcal{L}$ to the evaluation of a lift preserves the message. We first bound the last successive minimum of $\mathcal{L}$, then use rd independent lattice vectors to bound the canonical norm of a lift for every message.

Proposition 9. *For $\mathcal{L} = V^{-1} (I_r \otimes M_G) \mathbb{Z}^{rd}$, take successive minima in the maximum absolute coordinate norm on $\mathbb{C}^a \times \mathbb{R}^c$. Then*

$$\lambda_{rd}^{(\infty)}(\mathcal{L}) \leq \max_{0 \leq i < r} \max_{\substack{F(\xi)=0 \\ 0 \leq j < d}} |\xi|^i |\omega|^j |\omega - b|.$$

Moreover, every message m satisfies

$$\min_{\hat{m} \in \pi_b^{-1}(m)} \|\hat{m}\|_{\text{can}} \leq \frac{rd}{2} \lambda_{rd}^{(\infty)}(\mathcal{L}).$$

Proof. The polynomials $(B - b)Y^i B^j$, for $0 \leq i < r, 0 \leq j < d$, form a basis of $\mathfrak{J}_b$. The evaluations of $(B - b)Y^i B^j$ form a basis of $\mathcal{L}$, with coordinates

$$(\omega - b)\xi^i \omega^j.$$

The last successive minimum is at most the maximum norm of these basis vectors, proving the first bound.

Choose rd linearly independent vectors $v_1, \dots, v_{rd} \in \mathcal{L}$ with

$$\|v_\ell\|_\infty \leq \lambda_{rd}^{(\infty)}(\mathcal{L}).$$

For any $x = \sum_{\ell=1}^{rd} t_\ell v_\ell$ in the evaluation space, choose integers z_ℓ with $|t_\ell - z_\ell| \leq 1/2$. Then

$$\left\| x - \sum_{\ell=1}^{rd} z_\ell v_\ell \right\|_\infty \leq \frac{1}{2} \sum_{\ell=1}^{rd} \|v_\ell\|_\infty \leq \frac{rd}{2} \lambda_{rd}^{(\infty)}(\mathcal{L}).$$

Apply the inequality to the evaluation of an integer lift of m . Subtracting the lattice vector $\sum_\ell z_\ell v_\ell$ changes the lift by an element of $\mathfrak{I}_b$ and preserves m . The new lift satisfies the second bound. $\square$

Rounding in the explicit basis also gives the uniform lift bound

$$\max_{m \in \mathbb{F}_{p^r}} \min_{\hat{m} \in \pi_b^{-1}(m)} \|\hat{m}\|_{\text{can}} \leq \frac{1}{2} \|V^{-1}(I_r \otimes M_G)\|_{\infty \rightarrow \infty}. \quad (4)$$

Indeed, write the evaluation of any integer lift as $V^{-1}(I_r \otimes M_G)t$ with $t \in \mathbb{R}^{rd}$. Subtracting $V^{-1}(I_r \otimes M_G)[t]$ preserves the message, and $\|t - [t]\|_\infty \leq 1/2$ gives the bound. This is the same rounding construction as in Proposition 3, now measured in the canonical norm.

The same basis gives a bound on $\|V^{-1}(I_r \otimes M_G)\|_{\infty \rightarrow \infty}$. The row of $V^{-1}(I_r \otimes M_G)$ corresponding to (ξ, ω) is $((\omega - b)\xi^i \omega^j)_{i,j}$. The triangle inequality therefore gives

$$\|V^{-1}(I_r \otimes M_G)\|_{\infty \rightarrow \infty} \leq \max_{\substack{F(\xi)=0 \\ G(\omega)=0}} |\omega - b| \left(\sum_{i=0}^{r-1} |\xi|^i \right) \left(\sum_{j=0}^{d-1} |\omega|^j \right). \quad (5)$$

The domain uses the coefficient infinity norm, and the evaluation space uses complex absolute values. The maximum is taken after multiplying the factors at each root pair.

Substituting the successive-minimum lift bound into Proposition 7 gives an operator-norm bound for every message. Combining Propositions 7 and 9 with Lemma 2 gives the following bound in terms of the coefficients and degree gaps.

Theorem 1. *Suppose that $F(Y) = Y^r + \sum_{i=0}^{h_F} f_i Y^i$, $G(B) = B^d + \sum_{j=0}^{h_G} g_j B^j$ are squarefree, where $0 \leq h_F < r$ and $0 \leq h_G < d$. Write $\|F\|_1 = 1 + \sum_{i=0}^{h_F} |f_i|$ and $\|G\|_1 = 1 + \sum_{j=0}^{h_G} |g_j|$. For each $m \in \mathbb{F}_{p^r}$, one can choose an integer lift $\hat{m} \in \pi_b^{-1}(m)$ so that, for every integer $k \geq 1$ and messages $m_1, \dots, m_k$, the corresponding lifts satisfy*

$$\left\| \prod_{j=1}^k \hat{m}_j \right\|_{\text{op}} \leq \|V\|_{\infty \rightarrow \infty} \|V^{-1}\|_{\infty \rightarrow \infty} \left[\frac{rd}{2} \|F\|_1^{\frac{r-1}{\Delta_F}} \|G\|_1^{\frac{d-1}{\Delta_G}} \left(|b| + \|G\|_1^{\frac{1}{\Delta_G}} \right) \right]^k.$$

Proof. For each message, Proposition 9 gives an integer lift $\hat{m} \in \pi_b^{-1}(m)$ satisfying

$$\|\hat{m}\|_{\text{can}} \leq \frac{rd}{2} \lambda_{rd}^{(\infty)}(\mathcal{L}) \leq \frac{rd}{2} \cdot \max_{\substack{0 \leq i < r \\ 0 \leq j < d}} \max_{\substack{F(\xi)=0 \\ G(\omega)=0}} |\xi|^i |\omega|^j |\omega - b|.$$

Lemma 2 gives $|\xi| \leq \|F\|_1^{1/(\Delta_F)}$, $|\omega| \leq \|G\|_1^{1/(\Delta_G)}$. Substituting the root bounds and $|\omega - b| \leq |\omega| + |b|$ into the bound on each $\|\hat{m}_j\|_{\text{can}}$, then applying the product bound in Proposition 7, proves the theorem. $\square$

For numerical evaluation, (4) provides another choice of A in the product bound of Section 4.2.

4.4 Numerical Examples

Parameters and Lifts. For the secp256k1 prime $p = 2^{256} - 2^{32} - 977$, we use

$$F_1(Y) = Y, \quad F_2(Y) = Y^2 + 1, \quad F_4(Y) = Y^4 + 2Y - 1$$

and

$$\begin{aligned} G_{128}(B) &= B^{128} - B^{16} - B^5 + B^3 - B^2 - B + 3, \\ G_{256}(B) &= B^{256} - B^{32} - B^{10} + B^7 - B^6 - B^5 + B^3 + 2B^2 - B + 1. \end{aligned}$$

The polynomials F_2, F_4 are irreducible modulo p , and $G_{128}(4) = G_{256}(2) = p$. All the polynomials are squarefree. The computed G roots are nonreal, so each field element uses $rd/2$ CKKS slots for all six pairs (r, d) .

For each r , we use 32 messages sampled uniformly from $\mathbb{F}_{p^r}$. We lift each message by base- b decomposition. Appendix C gives the computed norms of V, V^{-1} , lattice and lift bounds, and multiplication norms for $k = 1, 3, 7$, together with the full numerical tables.

Maximum Multiplication Count. For the sequential product in Section 4.1, assume $\|\hat{m}_j\|_{\text{can}} \leq A$ and $\|e_j\|_{\infty} \leq \varepsilon$ for every fresh input, and $\|E_i\|_{\infty} \leq \varepsilon_i$ for each multiplication. Define

$$\begin{aligned} s_0 &= \|V^{-1}\|_{\infty \rightarrow \infty} \varepsilon, \\ s_i &= (A + \|V^{-1}\|_{\infty \rightarrow \infty} \varepsilon) s_{i-1} + A^i \|V^{-1}\|_{\infty \rightarrow \infty} \varepsilon + \|V^{-1}\|_{\infty \rightarrow \infty} \varepsilon_i. \end{aligned} \tag{6}$$

The identity

$$\nu_i - P_i = (\nu_{i-1} - P_{i-1})\mu_i + P_{i-1}e_i + E_i$$

and $\|P_{i-1}\|_{\text{can}} \leq A^i$ give, by induction,

$$\|\nu_i - P_i\|_{\text{can}} \leq s_i, \quad \|\nu_i - P_i\|_{\infty} \leq \|V\|_{\infty \rightarrow \infty} s_i.$$

Thus t successive multiplications satisfy the decoding condition at every step if $\|V\|_{\infty \rightarrow \infty} s_i < 1/2$ for all $0 \leq i \leq t$. The product contains $t + 1$ fresh inputs. When $\varepsilon_i = 0$ for every i , the recurrence gives

$$s_t = (A + \|V^{-1}\|_{\infty \rightarrow \infty} \varepsilon)^{t+1} - A^{t+1}.$$

Table 2 gives the largest such t for $\varepsilon = \varepsilon_i = 2^{-40}, 2^{-80}, 2^{-120}$. Both columns use the computed norms of V, V^{-1} . Bound uses $A = \frac{1}{2}\|V^{-1}(I_r \otimes M_G)\|_{\infty \rightarrow \infty}$ from (4), providing a lift for every message. Empirical takes $A = \max_{\text{sampled } m} \|\hat{m}\|_{\text{can}}$.

Table 2. Maximum multiplication counts t satisfying (6) and the decoding condition at every step. Each product contains $t + 1$ fresh inputs. Every input error bound ε and multiplication error bound ε_i equals the value above the columns. Bound uses the computed composed-operator bound in (4); Empirical uses the computed maximum canonical norm over the 32 sampled lifts. Both use computed norms of V, V^{-1} to evaluate the sufficient decoding condition under the stated error bounds.

r	d	$\varepsilon = 2^{-40}$		$\varepsilon = 2^{-80}$		$\varepsilon = 2^{-120}$	
		Bound	Empirical	Bound	Empirical	Bound	Empirical
1	128	3	4	7	9	11	15
1	256	2	4	7	10	11	16
2	128	2	3	6	9	10	14
2	256	2	3	6	9	10	15
4	128	2	2	5	7	8	12
4	256	2	2	5	7	8	12

Under these error assumptions, the sampled lifts for $r = 1$ permit 4, 9, 15 multiplications for $d = 128$ and 4, 10, 16 for $d = 256$, respectively. The CKKS parameters must satisfy the stated error bounds, and each multiplication also requires an available rescaling level.

5 Bootstrapping for Bi-Spectral Encoding

Multiplication in the bi-spectral encoding increases the carry I and error e , and rescaling decreases the ciphertext modulus. Correct decoding requires the error e to remain below the rounding threshold. Correct decryption also requires a ciphertext modulus large enough for the plaintext $m + I$. We propose a bootstrapping procedure that reduces the carry and error and restores the ciphertext modulus.

For $u \in \mathcal{P}$, let $\llbracket u \rrbracket \in \mathbb{R}^{rd}$ be the coefficient vector of u . Let $\text{Ecd}_{\text{ev}} : \mathcal{P} \rightarrow \mathcal{R}$ be the composition $\mathcal{P} \rightarrow \mathbb{C}^a \times \mathbb{R}^c \hookrightarrow \mathbb{C}^{N/2} \rightarrow \mathcal{R}$.

5.1 Refreshing the Carry

Let $q > 0$ be the base ciphertext modulus. Recall the coefficient lattice $\mathcal{L}_{\text{coeff}} = (I_r \otimes M_G)\mathbb{Z}^{rd}$ of the carry ideal $\mathfrak{J}_b$ from Section 3.1. A linear map from $\mathcal{L}_{\text{coeff}}$ onto $q\mathbb{Z}^{rd}$ makes each carry coefficient divisible by q . Reduction modulo q then removes the transformed carry in coefficient encoding. Define

$$L := \frac{1}{q} I_r \otimes M_G.$$

By Proposition 2, $I_r \otimes M_G$ is a basis matrix of $\mathcal{L}_{\text{coeff}}$, so L is invertible and

$$L(q\mathbb{Z}^{rd}) = \mathcal{L}_{\text{coeff}}, \quad L^{-1}(\mathcal{L}_{\text{coeff}}) = q\mathbb{Z}^{rd}.$$

The identical blocks and the row-sum bound from Proposition 3 give

$$q\|L\|_{\infty \rightarrow \infty} = \|M_G\|_{\infty \rightarrow \infty} \leq 1 + |b| + \max_{0 \leq k < d} |g_k|.$$

Equivalently, L is multiplication by $(B - b)/q$ on $\mathcal{P}$, written in coefficient coordinates.

Carry refresh consists of three steps.

1. l2C converts an encryption of $\text{Ecd}_{\text{ev}}(I + m + e)$ to an encryption of

$$\text{Ecd}_{\text{coeff}}(L^{-1}(\llbracket I + m + e \rrbracket)) = qJ + \text{Ecd}_{\text{coeff}}(L^{-1}(\llbracket m + e \rrbracket)),$$

where $qJ = \text{Ecd}_{\text{coeff}}(L^{-1}(\llbracket I \rrbracket))$ is a multiple of q .

2. Truncate reduces the ciphertext modulo q , and lifts the reduced ciphertext to modulus Q_{top} . Truncate gives an encryption of

$$qJ' + \text{Ecd}_{\text{coeff}}(L^{-1}(\llbracket m + e \rrbracket)) = \text{Ecd}_{\text{coeff}}(L^{-1}(\llbracket I' + m + e \rrbracket))$$

for $J' \in \mathcal{R}$ determined by the ModRaise operation.

3. C2l reverses the encoding conversion of l2C, converting an encryption of $\text{Ecd}_{\text{coeff}}(L^{-1}(\llbracket I' + m + e \rrbracket))$ to an encryption of $\text{Ecd}_{\text{ev}}(I' + m + e)$ at ciphertext modulus Q .

The ModRaise step in Truncate restores the ciphertext modulus.

Both l2C and C2l are compositions of three homomorphic linear maps, each evaluated as a plaintext-ciphertext matrix-vector product:

$$\text{l2C} = \text{S2C} \circ L^{-1} \circ \text{M2S}, \quad \text{C2l} = \text{S2M} \circ L \circ \text{C2S}.$$

M2S and S2M convert between Ecd_{ev} and slot encoding by homomorphically evaluating V and V^{-1} , respectively. The DFTs in S2C and C2S have factorizations used for homomorphic evaluation. The maps V , V^{-1} , L , and L^{-1} have no such factorization, so we evaluate the four maps by the baby-step giant-step method of Halevi and Shoup.

The following proposition bounds the canonical norm of the refreshed plaintext $I' + m + e$.

Proposition 10 (Carry bound). *Let s be the secret key. Suppose that the refreshed plaintext satisfies*

$$L^{-1}(\llbracket I' + m + e \rrbracket) = \llbracket \beta + \alpha s \rrbracket,$$

where the coefficients of β, α lie in $[-q/2, q/2)$. Here $\llbracket \beta + \alpha s \rrbracket$ denotes the vector of the rd coefficients used by coefficient encoding. Then

$$\|I' + m + e\|_{\text{can}} \leq \frac{1 + \|s\|_1}{2} \|V^{-1}(I_r \otimes M_G)\|_{\infty \rightarrow \infty}.$$

The composed operator norm is bounded in (5).

Proof. The coefficient identity gives

$$\|L^{-1}\llbracket I' + m + e \rrbracket\|_\infty \leq \|\beta\|_\infty + \|\alpha\|_\infty \|s\|_1 \leq \frac{q}{2}(1 + \|s\|_1).$$

Since $L = q^{-1}(I_r \otimes M_G)$, applying $V^{-1}L$ proves the bound. $\square$

For a sparse ternary secret of Hamming weight h , we have $\|s\|_1 = h$, so the bound becomes

$$\|I' + m + e\|_{\text{can}} \leq \frac{1+h}{2} \|V^{-1}(I_r \otimes M_G)\|_{\infty \rightarrow \infty}.$$

Sparse secret encapsulation can be applied to **Truncate** to reduce the effective Hamming weight in the bound.

5.2 Refreshing the Error

The carry refresh procedure gives $\llbracket I' + m + e \rrbracket$ in slot encoding. Since $\llbracket I' + m \rrbracket \in \mathbb{Z}^{rd}$ is integral and $\llbracket e \rrbracket$ is small, **EvalRound** can be used to reduce the error. Combining carry refresh and error refresh gives the following procedure.

1. We begin with an encryption of $\text{Ecd}_{\text{ev}}(I + m + e)$, and obtain $\text{Ecd}_{\text{slot}}(L^{-1}(\llbracket I' + m + e \rrbracket))$ after **I2C**, **Truncate**, and **C2S**.
2. Homomorphically apply L to obtain $\text{Ecd}_{\text{slot}}(\llbracket I' + m + e \rrbracket)$.
3. Apply **EvalRound** to obtain $\text{Ecd}_{\text{slot}}(\llbracket I' + m + e' \rrbracket)$, where $e' \ll e$ is the error left by **EvalRound**.
4. Apply **S2M** to obtain $\text{Ecd}_{\text{ev}}(I' + m + e')$.

EvalRound involves a polynomial approximation of $x \mapsto \lfloor x \rfloor = x - (x \bmod 1)$, whose degree depends on the size of the input domain. Applying L in the proof of Proposition 10, we obtain $\|I' + m + e\|_\infty \leq K_{\text{round}}$, where

$$K_{\text{round}} := \|M_G\|_{\infty \rightarrow \infty} \frac{1 + \|s\|_1}{2}.$$

The bound K_{round} exceeds the usual CKKS **ModRaise** bound by the factor $\|M_G\|_{\infty \rightarrow \infty}$. Since the depth of **EvalRound** grows logarithmically in K_{round} , **EvalRound** costs roughly $\log \|M_G\|_{\infty \rightarrow \infty}$ more levels than the usual CKKS **EvalMod**. Choosing G with small $\|M_G\|_{\infty \rightarrow \infty}$ keeps the additional cost practical.

6 Conclusion

We extend the finite-field encoding of Seuré and Suvanto to fields of large prime characteristic. Our bi-spectral encoding uses plaintext digit decomposition to obtain plaintexts with coefficient norm $O(p^{1/d})$ for suitable polynomials $F(Y), G(B)$, while supporting field addition and multiplication within a single CKKS ciphertext.

In our encoding, the supported multiplication count depends on the operator norms of plaintext products. We bound these norms in terms of the coefficient 1-norms and degree gaps of $F(Y)$ and $G(B)$. Together with bounds on the input and operation errors, this gives a sufficient condition on how many fresh plaintexts can be multiplied while ensuring correct decoding.

References

1. Alexandru, A., Kim, A., Polyakov, Y.: General functional bootstrapping using CKKS. Cryptology ePrint Archive, Paper 2024/1623 (2024), <https://eprint.iacr.org/2024/1623>
2. Bae, Y., Cheon, J.H., Kim, J., Stehlé, D.: Bootstrapping bits with CKKS. Cryptology ePrint Archive, Paper 2024/767 (2024), <https://eprint.iacr.org/2024/767>
3. Bae, Y., Kim, J., Stehlé, D., Suvanto, E.: Bootstrapping small integers with CKKS. In: Chung, K.M., Sasaki, Y. (eds.) *Advances in Cryptology – ASIACRYPT 2024, Part I. Lecture Notes in Computer Science*, vol. 15484, pp. 330–360. Springer (2024). https://doi.org/10.1007/978-981-96-0875-1_11
4. Brakerski, Z.: Fully homomorphic encryption without modulus switching from classical GapSVP. In: Safavi-Naini, R., Canetti, R. (eds.) *Advances in Cryptology – CRYPTO 2012. Lecture Notes in Computer Science*, vol. 7417, pp. 868–886. Springer (2012). https://doi.org/10.1007/978-3-642-32009-5_50
5. Brakerski, Z., Gentry, C., Vaikuntanathan, V.: Fully homomorphic encryption without bootstrapping. Cryptology ePrint Archive, Paper 2011/277 (2011), <https://eprint.iacr.org/2011/277>
6. Cha, G., Park, D., Lee, J.W.: Improved radix-based approximate homomorphic encryption for large integers via lightweight bootstrapped digit carry. Cryptology ePrint Archive, Paper 2025/1740 (2025), <https://eprint.iacr.org/2025/1740>
7. Cha, H., Hwang, I., Min, S., Seo, J., Song, Y.: MatriGear: Accelerating authenticated matrix triple generation with scalable prime fields via optimized HE packing. Cryptology ePrint Archive, Paper 2024/1502 (2024), <https://eprint.iacr.org/2024/1502>, published in IEEE Symposium on Security and Privacy, 2025
8. Chen, H., Laine, K., Player, R., Xia, Y.: High-precision arithmetic in homomorphic encryption. In: Smart, N.P. (ed.) *Topics in Cryptology – CT-RSA 2018. Lecture Notes in Computer Science*, vol. 10808, pp. 116–136. Springer (2018). https://doi.org/10.1007/978-3-319-76953-0_7
9. Cheon, J.H., Kim, A., Kim, M., Song, Y.: Homomorphic encryption for arithmetic of approximate numbers. In: *Advances in Cryptology – ASIACRYPT 2017. Lecture Notes in Computer Science*, vol. 10624, pp. 409–437. Springer (2017). https://doi.org/10.1007/978-3-319-70694-8_15
10. Drucker, N., Moshkovich, G., Pelleg, T., Shaul, H.: BLEACH: Cleaning errors in discrete computations over CKKS. *Journal of Cryptology* **37**, 3 (2024). <https://doi.org/10.1007/s00145-023-09483-1>
11. Fan, J., Vercauteren, F.: Somewhat practical fully homomorphic encryption. Cryptology ePrint Archive, Paper 2012/144 (2012), <https://eprint.iacr.org/2012/144>
12. Gama, M., Heydari Beni, E., Kang, J., Spiessens, J., Vercauteren, F.: Blind zk-SNARKs: for private proof delegation and verifiable computation over encrypted data. *IACR Communications in Cryptology* **2**(3) (2025). <https://doi.org/10.62056/akgyl8n4e>
13. Gao, M., Zheng, H.: FHE for SIMD arithmetic logic units with amortized $O(1)$ bootstrapping per ciphertext. Cryptology ePrint Archive, Paper 2026/233 (2026), <https://eprint.iacr.org/2026/233>
14. Geelen, R., Vercauteren, F.: Fully homomorphic encryption for cyclotomic prime moduli. Cryptology ePrint Archive, Paper 2024/1587 (2024), <https://eprint.iacr.org/2024/1587>
15. Kim, J.: Faster homomorphic integer computer. Cryptology ePrint Archive, Paper 2025/1440 (2025), <https://eprint.iacr.org/2025/1440>

16. Liu, J., Ma, S., Wang, A., Wang, X.: High-precision homomorphic ALU over arbitrary moduli with $O(1)$ bootstrapping. Cryptology ePrint Archive, Paper 2026/1869 (2026), <https://eprint.iacr.org/2026/1869>, version received 2 September 2026
17. Moody, D., Peralta, R., Perlner, R., Regenscheid, A., Roginsky, A., Chen, L.: Report on pairing-based cryptography. Journal of Research of the National Institute of Standards and Technology **120** (2015). <https://doi.org/10.6028/jres.120.002>
18. Peikert, C., Zarchy, D., Zyskind, G.: High-precision exact FHE made simple, general, and fast. Cryptology ePrint Archive, Paper 2025/2321 (2025), <https://eprint.iacr.org/2025/2321>, major revision of an IACR publication in CRYPTO 2026; third revision, 14 August 2026
19. Seur , T., Suvanto, E.: Finite-field arithmetic in CKKS. Cryptology ePrint Archive, Paper 2026/1102 (2026), <https://eprint.iacr.org/2026/1102>
20. Shen, X., Li, Z., Wang, R., Lu, X.: Arithmetic-to-boolean conversion in ALU with $O(1)$ bootstrapping via overflow cancellation. Cryptology ePrint Archive, Paper 2026/1840 (2026), <https://eprint.iacr.org/2026/1840>
21. Wei, Y., Ma, K., Gao, M., Zheng, H.: FHE for arbitrary modulus ALU and application to one-round threshold ECDSA. Cryptology ePrint Archive, Paper 2026/1836 (2026), <https://eprint.iacr.org/2026/1836>

A Matrix Representation of Root Evaluation

Use the coefficient basis $(Y^i B^j)_{i,j}$ with j varying first. List the selected root pairs with the c real pairs first, followed by one pair from each of the a nonreal conjugate pairs. Splitting the complex coordinates into real and imaginary parts gives the real matrix

$$V^{-1} = \begin{pmatrix} 1 \cdots \xi_1^i \omega_1^j & \cdots & \xi_1^{r-1} \omega_1^{d-1} \\ \vdots & \vdots & \vdots \\ 1 \cdots \xi_c^i \omega_c^j & \cdots & \xi_c^{r-1} \omega_c^{d-1} \\ 1 \cdots \Re(\xi_{c+1}^i \omega_{c+1}^j) & \cdots & \Re(\xi_{c+1}^{r-1} \omega_{c+1}^{d-1}) \\ 0 \cdots \Im(\xi_{c+1}^i \omega_{c+1}^j) & \cdots & \Im(\xi_{c+1}^{r-1} \omega_{c+1}^{d-1}) \\ \vdots & \vdots & \vdots \\ 1 \cdots \Re(\xi_{c+a}^i \omega_{c+a}^j) & \cdots & \Re(\xi_{c+a}^{r-1} \omega_{c+a}^{d-1}) \\ 0 \cdots \Im(\xi_{c+a}^i \omega_{c+a}^j) & \cdots & \Im(\xi_{c+a}^{r-1} \omega_{c+a}^{d-1}) \end{pmatrix}.$$

The column indexed by (i, j) corresponds to $Y^i B^j$. Both V^{-1} and its inverse V have size $rd \times rd$.

The canonical norm in Appendix B uses complex absolute values. In the real coordinates of the matrix representation, the unit ball is

$$|z_k| \leq 1 \quad (1 \leq k \leq c), \quad z_{c+2k-1}^2 + z_{c+2k}^2 \leq 1 \quad (1 \leq k \leq a).$$

For the entries $V_{\ell k}$ of the inverse matrix, the operator norm is

$$\|V\|_{\infty \rightarrow \infty} = \max_{1 \leq \ell \leq rd} \left(\sum_{k=1}^c |V_{\ell k}| + \sum_{k=1}^a \sqrt{V_{\ell, c+2k-1}^2 + V_{\ell, c+2k}^2} \right).$$

Each real coordinate contributes the absolute value of its matrix entry. Each complex coordinate contributes the Euclidean norm of the corresponding two entries, since its real and imaginary parts range over the unit disk.

B Errors Introduced by CKKS Operations

We convert CKKS errors in the cyclotomic ring to coefficient errors in $\mathcal{P}$ to apply the decoding condition. The norm of V bounds the error after recovering plaintext coefficients. The norm of V^{-1} bounds the evaluation coordinates supplied to CKKS. We use the coefficient infinity norm on $\mathcal{P}$ and the maximum complex absolute value on $\mathbb{C}^a \times \mathbb{R}^c$. Appendix A gives the real matrix representation.

For a cyclotomic coefficient error $\eta(X)$ at scale Δ , evaluation at a root ζ of $X^N + 1$, with $\zeta^N = -1$, gives

$$|\Delta^{-1}\eta(\zeta)| \leq N\Delta^{-1}\|\eta\|_\infty,$$

because $|\zeta| = 1$ and $\deg \eta < N$. The polynomial η represents the error before division by the scale Δ at the operation. Selecting the encoding coordinates and taking real parts at real root pairs cannot increase the maximum absolute coordinate of the evaluated error. Hence the coefficient error $E \in \mathbb{R}[Y, B]/(F(Y), G(B))$ satisfies

$$\|E\|_\infty \leq \|V\|_{\infty \rightarrow \infty} N\Delta^{-1}\|\eta\|_\infty.$$

Substituting (3) gives a bound depending on F, G . For an existing coefficient error e , the decoding condition in Section 3.2 holds whenever $\|e\|_\infty + \|E\|_\infty < 1/2$.

C Numerical Details

The following tables give the polynomial parameters, operator norms, lattice bounds, and sampled lifts used in Section 4.4.

Choice of Polynomials. We evaluate the preceding bounds for the secp256k1 prime

$$p = 2^{256} - 2^{32} - 977.$$

We use extension degrees $r = 1, 2, 4$, with

$$F_1(Y) = Y, \quad F_2(Y) = Y^2 + 1, \quad F_4(Y) = Y^4 + 2Y - 1.$$

The polynomials F_2 and F_4 are irreducible modulo p . The three choices of F have $(\|F\|_1, \Delta_F) = (1, 1), (2, 2), (4, 3)$, respectively. For the decomposition polynomial, we use

$$\begin{aligned} G_{128}(B) &= B^{128} - B^{16} - B^5 + B^3 - B^2 - B + 3, \\ G_{256}(B) &= B^{256} - B^{32} - B^{10} + B^7 - B^6 - B^5 + B^3 + 2B^2 - B + 1. \end{aligned}$$

Then $G_{128}(4) = G_{256}(2) = p$. Both polynomials are squarefree. Table 3 lists their coefficient norms and degree gaps.

Each pair (F_r, G_d) represents $\mathbb{F}_{p^r}$. The comparisons depend on the choice of F_r, G_d as well as r, d .

Table 3. Parameters of G_{128} and G_{256} : the evaluation base b , coefficient 1-norm, gap between the leading and next nonzero degrees, and coefficient matrix norm of multiplication by $B - b$.

G	b	$\ G\ _1$	Δ_G	$\ M_G\ _{\infty \rightarrow \infty}$
G_{128}	4	9	112	7
G_{256}	2	11	224	5

Evaluation and Interpolation Norms. We first compare the bounds on $\|V\|_{\infty \rightarrow \infty}$ and $\|V^{-1}\|_{\infty \rightarrow \infty}$. Table 4 separates the coefficient bound in Corollary 1 from the root-sum bound in Lemma 1. For the root pairs of the chosen F_r, G_d , the interpolation bound in Proposition 8 equals $\|V\|_{\infty \rightarrow \infty}$.

Table 4. Evaluation and interpolation norms. Coeff. Bound uses the coefficients and degree gaps of F, G ; Root Bound uses the root-sum formula evaluated at the computed roots. Computed gives numerical operator norms with complex absolute values in the evaluation space.

r	d	$\ V^{-1}\ _{\infty \rightarrow \infty}$			$\ V\ _{\infty \rightarrow \infty}$
		Coeff. Bound	Root Bound	Computed	Computed=Bound
1	128	571.31	398.89	256.08	1.17653
1	256	1346.73	742.63	474.80	1.84950
2	128	1379.26	797.78	509.87	1.17653
2	256	3251.28	1485.26	947.39	1.84950
4	128	5203.05	2815.76	1807.64	1.45489
4	256	12264.95	5242.20	3351.59	2.28707

Lattice and Lift Bounds. Next, Proposition 9 bounds $\lambda_{rd}^{(\infty)}(\mathcal{L})$ by

$$\max_{i,j,\xi,\omega} |\xi|^i |\omega|^j |\omega - b|.$$

Table 5 evaluates this bound using either the Cauchy bound in Lemma 2 or the computed roots. The Reduced Basis Bound is the maximum vector norm of a basis obtained by integer column operations on the two tensor factors of $V^{-1}(I_r \otimes M_G)$. We add or subtract one or two copies of another column whenever the operation decreases the column's canonical norm. We make at most 12 passes over the columns. Integer column operations preserve the lattice basis. The maximum norm of the resulting basis vectors therefore bounds $\lambda_{rd}^{(\infty)}(\mathcal{L})$, as in Proposition 9. The second inequality in Proposition 9 gives the Coeff. Bound and Root Bound for the worst-case message by multiplying the successive-minimum bounds by $rd/2$. We also compute the basis-rounding bound in (4) from Section 4.3. The operator norm uses real coordinates in the domain and complex absolute values

Table 5. Upper bounds on the last successive minimum and on the worst-case minimum canonical norm over all messages. Root Bound evaluates the first inequality in Proposition 9 at the computed roots; Coeff. Bound applies Lemma 2 to the same inequality. Reduced Basis Bound is the computed maximum vector norm after integer column operations on the evaluated basis. For the message bound, Coeff. Bound and Root Bound use the second inequality in Proposition 9; Computed Bound numerically evaluates the right-hand side of (4).

r	d	$\lambda_{rd}^{(\infty)}(\mathcal{L})$			$\max_{m \in \mathbb{F}_{p^r}} \min_{\hat{m} \in \pi_b^{-1}(m)} \ \hat{m}\ _{\text{can}}$		
		Coeff. Bound	Root Bound	Reduced Basis Bound	Coeff. Bound	Root Bound	Computed Bound
1	128	60.6360	30.4046	24.3155	3880.70	1945.89	558.06
1	256	46.1520	18.0865	13.9677	5907.46	2315.07	688.45
2	128	85.7522	30.4046	24.3155	10976.28	3891.78	1108.83
2	256	65.2688	18.0865	13.9677	16708.81	4630.15	1367.38
4	128	242.5439	82.5992	43.8995	62091.23	21145.40	3939.32
4	256	184.6080	49.1352	25.2174	94519.29	25157.20	4859.72

Table 6. Maximum coefficient and computed canonical norms over 32 uniformly sampled messages for each r , lifted by base- b decomposition. The last two columns give the slots per field element and the field elements per ciphertext with 32,768 CKKS slots.

r	d	$\max_{\text{sampled } m} \ \hat{m}\ _{\infty}$	$\max_{\text{sampled } m} \ \hat{m}\ _{\text{can}}$	Slots/field	Fields/ct
1	128	4	106.8715	64	512
1	256	2	79.2395	128	256
2	128	4	138.4751	128	256
2	256	2	107.9954	256	128
4	128	4	379.4721	256	128
4	256	2	295.8400	512	64

in the evaluation space. The successive minimum and the minimum over lifts are bounded using basis vectors and coordinate rounding.

Sampled Integer Lifts and Their Powers. For each r , we use 32 messages sampled uniformly from $\mathbb{F}_{p^r}$. For each sampled message m , construct a lift $\hat{m}$ as follows. Starting from each integer coefficient, choose the coefficient of B^j in $[-b/2, b/2)$ congruent to the remaining integer modulo b , subtract the coefficient, and divide by b . After $d-1$ steps, use the remaining integer as the final coefficient. The resulting $\hat{m}(Y, B)$ evaluates to $m(Y)$ at $B = b$. Table 6 gives the maximum norms over the sampled messages and the slot counts. The computed G roots are nonreal, giving $c = 0$ and $a = rd/2$ for all six pairs. With one coordinate from each simultaneous conjugate pair, a field element uses $rd/2$ slots. The last column uses 32,768 CKKS slots and $rd/2$ slots per field element. Table 7 compares Proposition 7 with the row norm of the coefficient matrix of multiplication by $\hat{m}^k$, taking the maximum of each quantity over the sampled messages.

Table 7. Bounds and computed values of $\max_{\text{sampled } m} \|\widehat{m}^k\|_{\text{op}}$ for the sampled lifts in Table 6. Each column takes the maximum over the same 32 messages. Lower and upper bounds use Proposition 7; Computed norm uses the maximum absolute row sum of each coefficient multiplication matrix.

r	d	k	Lower bound	Computed norm	Upper bound
1	128	1	$1.06872 \cdot 10^2$	$5.47000 \cdot 10^2$	$3.21985 \cdot 10^4$
		3	$1.22063 \cdot 10^6$	$3.82900 \cdot 10^6$	$3.67755 \cdot 10^8$
		7	$1.59233 \cdot 10^{14}$	$6.38846 \cdot 10^{14}$	$4.79741 \cdot 10^{16}$
1	256	1	$7.92395 \cdot 10^1$	$5.68000 \cdot 10^2$	$6.95832 \cdot 10^4$
		3	$4.97536 \cdot 10^5$	$2.88231 \cdot 10^6$	$4.36906 \cdot 10^8$
		7	$1.96151 \cdot 10^{13}$	$1.08379 \cdot 10^{14}$	$1.72248 \cdot 10^{16}$
2	128	1	$1.38475 \cdot 10^2$	$9.95000 \cdot 10^2$	$8.30681 \cdot 10^4$
		3	$2.65531 \cdot 10^6$	$1.02374 \cdot 10^7$	$1.59286 \cdot 10^9$
		7	$9.76342 \cdot 10^{14}$	$3.36339 \cdot 10^{15}$	$5.85686 \cdot 10^{17}$
2	256	1	$1.07995 \cdot 10^2$	$1.01800 \cdot 10^3$	$1.89229 \cdot 10^5$
		3	$1.25955 \cdot 10^6$	$8.36016 \cdot 10^6$	$2.20698 \cdot 10^9$
		7	$1.71332 \cdot 10^{14}$	$8.39199 \cdot 10^{14}$	$3.00206 \cdot 10^{17}$
4	128	1	$3.79472 \cdot 10^2$	$3.71100 \cdot 10^3$	$9.97979 \cdot 10^5$
		3	$5.46436 \cdot 10^7$	$3.38154 \cdot 10^8$	$1.43708 \cdot 10^{11}$
		7	$1.13308 \cdot 10^{18}$	$6.22658 \cdot 10^{18}$	$2.97989 \cdot 10^{21}$
4	256	1	$2.95840 \cdot 10^2$	$3.11200 \cdot 10^3$	$2.26771 \cdot 10^6$
		3	$2.58923 \cdot 10^7$	$1.20945 \cdot 10^8$	$1.98473 \cdot 10^{11}$
		7	$1.98335 \cdot 10^{17}$	$8.89661 \cdot 10^{17}$	$1.52030 \cdot 10^{21}$