HERatio: Homomorphic Encryption of Rationals using Laurent polynomials

Luke Harmon, Gaetan Delavignette, and Hanes Oliveira

Algemetric Inc.
{lharmon,gdelavignette,holiveira}@algemetric.com

Abstract. In this work we present HERatio, a homomorphic encryption scheme that builds on the scheme of Brakerski, and Fan and Vercauteren. Our scheme naturally accepts Laurent polynomials as inputs, allowing it to work with rationals via their bounded base-b expansions. This eliminates the need for a specialized encoder and streamlines encryption, while maintaining comparable efficiency to BFV. To achieve this, we introduce a new variant of the *Polynomial Learning With Errors* (PLWE) problem which employs Laurent polynomials instead of the usual "classic" polynomials, and provide a reduction to the PLWE problem.

Keywords: homomorphic encryption · Laurent polynomials · rational numbers · polynomial learning with errors.

1 Introduction

A large part of current research and development in Homomorphic Encryption (HE) is focused on efficient implementation with suitable software and/or hardware support and developing practically usable libraries for HE that can possibly be used for different machine learning and data analysis applications. These works clearly aim towards making HE practical for real-world applications.

The state-of-the-art HE schemes are defined to process (modulo) integer inputs or polynomial inputs (with modulo integer coefficients). For a significantly large number of practical applications, an HE scheme should be able to operate on real/rational numbers. In any practical HE an important issue is to convert the application data (type) to the data type suitable for the HE. This is usually achieved by encoding real-valued data to convert it into a "suitable" form compatible with homomorphic encryption. Any encoding must come with a matching decoding. Additionally, such an encoding must be homomorphic with respect to addition and multiplication, and injective. Most importantly, any such encoding technique must be efficient and not hinder the efficiency of the underlying HE scheme.

The interest in HE-compatible encodings to process real/rational inputs efficiently is evident from a number of previous works e.g. [2, 3, 11, 16, 21].

In most of the RLWE (Ring Learning with Error) hardness-based homomorphic encryption schemes a plaintext is viewed as an element of the ring $R_t = \mathbb{Z}_t[x]/\Phi_m\mathbb{Z}_t[x]$ where $\Phi_m(x)$ is the m-th cyclotomic polynomial and $\mathbb{Z}_t$ is

the ring of integers modulo t. Encoding integer inputs to a polynomial in R_t is relatively straightforward, namely one can consider the base t representation of the integer. For allowing integer and rational inputs one must define an encoding converting elements of $\mathbb{Z}$ or $\mathbb{Q}$ (typically represented as fixed-point decimal numbers in applications) into elements of R_t. Previous works [4,9,10,12,13,18,22,25] have proposed several encoding methods for integers and rationals. One previously taken approach is to scale the fixed-point numbers to integers and then encode them as polynomials (using a suitable base). Another approach is to consider, them as fractional numbers. In [12] it was shown that these two representations are isomorphic. As pointed out in [12] the latter approach, although avoiding the overhead of bookkeeping with homomorphic ciphertexts, is difficult to analyse.

All of these aforementioned encodings share a problem (discussed in [9, 12]) namely, t must have sufficiently large value for the encoding to work correctly. This large value of t results in faster noise growth and consequently one may need to choose large parameter for the overall homomorphic encryption scheme hindering the efficiency. A clever solution to this problem was proposed by Chen, Len, Player and Xia [9], which borrows a mathematical technique from Hoffstein and Silverman [20] and combines it with the homomorphic encryption scheme proposed by Fan and Vercauteren [17]. The main idea of the encoding in [9] is to replace the modulus t with the polynomial $x - b$ for some positive integer b and turning the plaintext space into the quotient ring $\mathbb{Z}/(b^n + 1)\mathbb{Z}$. Another solution to this problem introduced by Castryck *et al* in [8] encodes rationals by computing their base-b expansion, replacing b by an unknown x, and then mapping the resulting Laurent polynomial to an appropriate "classic" polynomial using a novel ring homomorphism. Similar encodings have been considered in [3, 10, 14, 16].

Our Results We introduce here a homomorphic encryption scheme for rationals. HERatio naturally accepts Laurent polynomials corresponding to bounded base-b expansions of rational numbers without the need of a specialized encoder. While enjoying efficiency comparable to BFV, it is more mathematically streamlined than prior art, and also mitigates the difficulty in choosing parameters to make sure a rational encoding "plays well" with the underlying HE scheme. HERatio may be viewed as a variant of the well-known Brakerski/Fan-Vercauteren (BFV) scheme [5,17], and is obtained (among other modifications) by replacing the rings of "classic" polynomials in BFV by rings of Laurent polynomials. Of course, it must be shown that these changes do not disturb the security of BFV. This is done by introducing a new hardness assumption using Laurent polynomials that can be reduced to the hardness assumption used by BFV. In particular, we introduce a new (decisional) version of the *Polynomial Learning With Errors* (PLWE) problem which uses the Laurent polynomial ring $\mathbb{Z}_q[x^{\pm 1}]/f\mathbb{Z}_q[x^{\pm 1}]$ instead of $\mathbb{Z}_q[x]/f\mathbb{Z}_q[x]$ as in the decisional-PLWE problem [7, 15, 19, 26]. We then use the novel encoding homomorphism from [8] to show that the new problem based on Laurent polynomials is at least as hard as the decisional-PLWE prob-

lem under certain conditions, and that modifying BFV scheme to use the new problem results in comparable efficiency.

2 Notations and Foundations

2.1 Notations

$\mathbb{Z}$ will denote the ring of integers, and $\mathbb{Z}_a$ will denote the quotient ring $\mathbb{Z}/a\mathbb{Z}$. For $a \in \mathbb{Z}$, we will identify the elements of $\mathbb{Z}_a$ with integer representatives $\big[-\lfloor (a-1)/2 \rfloor, \lceil (a-1)/2 \rceil\big] \cap \mathbb{Z}$. For a ring R, $R[x]$ will denote the ring of polynomials in x with coefficients from R, and $R[x^{\pm 1}]$ will denote the ring of Laurent polynomials. For non-negative integers ℓ, k we use $\mathbb{Z}[x^{\pm 1}]^k_{-\ell}$ $\big($resp. $\mathbb{Z}_a[x^{\pm 1}]^k_{-\ell}\big)$ to denote the subset of $\mathbb{Z}[x^{\pm 1}]$ $\big($resp. $\mathbb{Z}_a[x^{\pm 1}]\big)$ with exponents ranging from $-\ell$ to k. For n a power of 2, Φ_{2n} denotes the $2n^{\text{th}}$ cyclotomic polynomial $x^n + 1$. For a distribution χ over a set A and a function $f : A \to A'$, we denote by $f(\chi)$ the distribution over A' induced by χ and f, $x \leftarrow \chi$ will mean that x is chosen from A according to the distribution χ.

2.2 Polynomial Learning With Errors

We first recall the *Polynomial Learning With Errors* (PLWE) problem [7,24], on which the well-known Brakerski/Fan-Vercauteren (BFV) scheme is based.

Definition 1 (Decisional PLWE). *For all $\kappa \in \mathbb{N}$, let $f(x) = f_\kappa(x)$ be a polynomial of degree $n = n(\kappa)$, and let $q = q(\kappa)$ be a prime integer. Let $R = \mathbb{Z}[x]/f\mathbb{Z}[x]$, $R_q = R/qR$, and χ denote a distribution over R. The decisional-PLWE problem $\mathsf{PLWE}_{f,q,\chi}$ is, for any $\ell = poly(\kappa)$, to distinguish the sets*

$$\{(a_i, a_i \cdot s + e_i)\}_{i \in [\ell]} \text{ and } \{(a_i, u_i)\}_{i \in [\ell]}$$

where s is sampled from the distribution χ, the a_i are uniform in R_q, the error polynomials e_i are sampled from χ, and the ring elements u_i are uniformly random over R_q.

$\mathsf{PLWE}_{f,q,\chi}$ is hard for well-chosen parameters. It is also worth noting that for noise growth and performance reasons, it is possible to use a variant in which the coefficients of the secret key are uniformly selected from $\big\{-1, 0, 1\big\}$. This was originally suggested as an optimization in [17]. It was also shown in [6] that certain small-secret PLWE variants are as hard as those with $s \leftarrow \chi$ if the degree is sufficiently increased, even though more attacks can be used in this scenario, as shown in [1].

2.3 The BFV Scheme

Since our scheme is a variant of the Brakerski/Fan-Vercauteren scheme, we briefly recall some of the relevant details.

For its security, BFV relies on the hardness of the decisional-PLWE problem with $f(x) = \Phi_{2n}(x)$, and χ a discrete gaussian distribution on R with small standard deviation, normally chosen to be around 3.2 in practice [23].

The following algorithms are the basis of a common variant of the BFV scheme using the ternary distribution for s and u. Let $\Delta = \lfloor q/t \rfloor$ such that $q = \Delta t + r_t(q)$ for some $r_t(q) < t$. It should be assumed that $t \ll q$, which is required for most useful parameters.

BFV.SecretKeyGen: Sample $s \in R$ with coefficients uniformly distributed in $\{-1, 0, 1\}$.
Output

$$\mathsf{sk} = s$$

BFV.PublicKeyGen(sk): Let $s = \mathsf{sk}$. Sample $a \leftarrow R_q$, and $e \leftarrow \chi$.
Output

$$\mathsf{pk} = \Big(\big[-(as + e) \big]_q, a \Big) \in R_q \times R_q$$

BFV.Enc($\mathsf{pk}, m \in R_t$): To encrypt a message $m \in R_t$. Let $\mathsf{pk} = (p_0, p_1)$. Sample $u \in R$ with coefficients uniform in $\{-1, 0, 1\}$, and $e_0, e_1 \leftarrow \chi$.
Output

$$\mathsf{ct} = \big([\Delta m + p_0 u + e_0]_q, [p_1 u + e_1]_q \big) \in R_q \times R_q$$

BFV.Dec($\mathsf{sk}, \mathsf{ct} \in R_q \times R_q$). Let $s = \mathsf{sk}$ and $\mathsf{ct} = (c_0, c_1)$.
Output

$$m'(X) = \left[\left\lfloor \frac{t}{q} [c_0 + c_1 s]_q \right\rceil \right]_t \in R_t$$

The remaining protocols, as well as a proof of correctness can be found in [17]. The security is proven in [24] through an indistinguishability argument which relies on the hardness of the decisional-PLWE problem (Definition 1).

3 LWE with Laurent Polynomials

Let $f \in \mathbb{Z}[x]$ and k, ℓ be non-negative integers such that $k + \ell + 1 = \deg f$. Here we introduce a new (decisional) version of the LWE problem which uses the ring $\mathbb{Z}[x^{\pm 1}]/f\mathbb{Z}[x^{\pm 1}]$ with representatives $\mathbb{Z}[x^{\pm 1}]^k_{-\ell}$ instead of $\mathbb{Z}[x]/f\mathbb{Z}[x]$ with representatives $\mathbb{Z}[x^{\pm 1}]^{k+\ell}_0 = \{\mathsf{p}(x) \in \mathbb{Z}[x] | \deg \mathsf{p} < \deg f\}$ (as in the decisional-PLWE problem). We then show that the new problem based on Laurent polynomials is at least as hard as the decisional-PLWE problem under certain conditions. Throughout this section, $\mathfrak{R} = \mathbb{Z}[x]$ and $\mathfrak{L} = \mathbb{Z}[x^{\pm 1}]$. Also, for $a \in \mathbb{Z}$, $\mathfrak{R}_a = \mathfrak{R}/a\mathfrak{R}$ and $\mathfrak{L}_a = \mathfrak{L}/a\mathfrak{L}$.

3.1 From "classic" polynomials to Laurent polynomials

Before introducing the new problem, we build the tools required to show that the new problem is at least as hard as $\mathsf{PLWE}_{f,q,\chi}$ in certain cases. We first show that the rings $\mathbb{Z}[x]/f\mathbb{Z}[x]$ and $\mathbb{Z}[x^{\pm 1}]/f\mathbb{Z}[x^{\pm 1}]$ are isomorphic for certain polynomials f. The isomorphism ends up simply being the map $\mathsf{p}(x)+f\mathbb{Z}[x] \mapsto \mathsf{p}(x)+f\mathbb{Z}[x^{\pm 1}]$, which means that if we are to use proper Laurent polynomials as reprentatives, we need a way to switch representatives in the ring $\mathbb{Z}[x^{\pm 1}]/f\mathbb{Z}[x^{\pm 1}]$.

Recall the following classic theorem from elementary algebra.

Lemma 1 (Second Isomorphism Theorem). *Let R be a ring, S a subring of R, and I an ideal of R. Then $S + I$ is a subring of R, $S \cap I$, and*

$$\varphi : (S + I)/I \to S/(S \cap I) \text{ defined by } x + I \mapsto x + S \cap I$$

is a ring isomorphism.

Proposition 1. *Let $f \in \mathfrak{R}$ with $f(0) \in \mathbb{Z}$ a unit, $L = \mathfrak{L}/f\mathfrak{L}$, and $R = \mathfrak{R}/f\mathfrak{R}$. Then there is a ring isomorphism $L \cong R$.*

Proof. $\mathfrak{R}$ is a subring of $\mathfrak{L}$, and $f\mathfrak{L}$ is an ideal of $\mathfrak{L}$. So by lemma 1, $(\mathfrak{R}+f\mathfrak{L})/f\mathfrak{L} \cong \mathfrak{R}/(\mathfrak{R} \cap f\mathfrak{L})$. We claim that $\mathfrak{R} + f\mathfrak{L} = \mathfrak{L}$. That the sum is contained in $\mathfrak{L}$ is easy. For the other containment it suffices to show that $x^k \in \mathfrak{R} + f\mathfrak{L}$ for all $k \in \mathbb{Z}$. That this holds for $k \geq 0$ is immediate from the definition of $\mathfrak{R}$. To see that this also holds for negative powers, first observe that $x^{-1}(f(x) - f(0)) \in \mathfrak{R}$. Now, $f(0)x^{-1} = x^{-1}(f(0) - f(x)) + x^{-1}f(x) \in \mathfrak{R} + f\mathfrak{L}$. Whence $x^{-1} = f(0)^{-1}\big(f(0)x^{-1}\big) \in \mathfrak{R} + f\mathfrak{L}$, since $f(0)$ is a unit. An easy induction then shows that $x^k \in \mathfrak{R} + f\mathfrak{L}$ for all $k < 0$. Clearly $\mathfrak{R} \cap f\mathfrak{L} = f\mathfrak{R}$, whence $\mathfrak{L}/f\mathfrak{L} \cong \mathfrak{R}/f\mathfrak{R}$. Equivalently, $L \cong R$, as desired.

Remark 1. The same result holds if we replace everywhere $\mathbb{Z}$ by $\mathbb{Z}_a$, $a \in \mathbb{Z}$, but with the slightly better condition that $f(0)$ is invertible modulo a.

As mentioned at the beginning of this subsection, we need to map representatives in the set $\mathbb{Z}_a[x^{\pm 1}]^k_{-\ell}$ to their equivalents in $\mathbb{Z}_a[x^{\pm 1}]^{k+\ell}_0$ and vice versa. This can be done efficiently (using only matrix multiplications and inversions) using a ring homomorphism introduced by Castryck *et al* in [8]. We pause briefly to describe this homomorphism and give an example.
Let $f(x) \in \mathfrak{R}_a$ such that $f(0) \in \mathbb{Z}_a$ is a unit. The ring homomorphism $\mathfrak{L}_a \to R_a = \mathfrak{R}_a/f\mathfrak{R}_a$ is induced by the correspondences

$$x \mapsto x \quad \text{and} \quad x^{-1} \mapsto -g(x)f(0)^{-1}, \quad \text{where} \quad f(x) = g(x)x + f(0). \tag{1}$$

Let ℓ, k be non-negative integers satisfying $\deg f = k + \ell + 1$. The ring homomorphism in Equation (1) induces a free $\mathbb{Z}_a$-module isomorphism $\eta_{f,(-\ell,k)} : \mathbb{Z}_a[x^{\pm 1}]^k_{-\ell} \to \mathbb{Z}_a[x^{\pm 1}]^{k+\ell}_0$. As a free module isomorphism, $\eta_{f,(-\ell,k)}$ can be computed efficiently using a matrix multiplication. If the polynomial f and integers k, ℓ are clear from context, we will simply write η.

Example 1. Consider $f(x) = x^4 - x^3 + x^2 + x + 1 \in \mathbb{Z}_3[x]$. $f(0) \in \mathbb{Z}_3$ is a unit and $f(x) = x(x^3 - x^2 + x + 1) + 1$, so η_f is defined by

$$x \mapsto x \quad \text{and} \quad x^{-1} \mapsto -x^3 + x^2 - x - 1$$

Taking $\ell = 3$ and $k = 0$ makes the domain of η_f $\mathbb{Z}_3[x^{\pm 1}]_{-3}^0 = \{ax^{-3} + bx^{-2} + cx^{-1} + d \,|\, a, b, c, d \in \mathbb{Z}_3\}$, and the range $\mathbb{Z}_3[x^{\pm 1}]_0^3 = \{a + bx + cx^2 + dx^3 \,|\, a, b, c, d \in \mathbb{Z}_3\}$. The domain and range being free $\mathbb{Z}_3$-modules allows us to represent η_f (encoding map) and its inverse (decoding map) using matrices:

$$\text{encode matrix} = \begin{bmatrix} -1 & 0 & -1 & 1 \\ 1 & -1 & -1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & -1 & 0 \end{bmatrix}, \qquad \text{decode matrix} = \begin{bmatrix} 0 & -1 & -1 & 0 \\ 0 & -1 & 1 & -1 \\ 0 & -1 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{bmatrix}.$$

It is worth mentioning, as noted in [8], that when $f(x) = \Phi_{2n}(x) = x^n + 1$ (a common choice for PLWE) η_f can be computed without any matrix multiplications. In particular, η_f is defined by $x \mapsto x$ and $x^{-1} \mapsto -x^{n-1}$, meaning that a simple "degree shift" may be applied to any negative powers in a Laurent polynomial argument to compute η_f.

Proposition 2. *Let $f(x) = x \cdot g(x) + f(0) \in \mathbb{Z}_a[x]$ such that $f(0)$ is a unit, and $\ell, k \in \mathbb{Z}$ be non-negative. If $\eta_{f,(-\ell,k)} : \mathbb{Z}_a[x^{\pm 1}]_{-\ell}^k \to \mathbb{Z}_a[x^{\pm 1}]_0^{k+\ell}$ is defined by $x \mapsto x$ and $x^{-1} \mapsto -g(x)f(0)^{-1}$, then $\alpha(x) - \eta_{f,(-\ell,k)}\big(\alpha(x)\big) = 0 \bmod f(x)$ for all $\alpha \in \mathbb{Z}_a[x^{\pm 1}]_{-\ell}^k$.*

Proof. Let $\alpha = a_{-\ell}x^{-\ell} + \cdots + a_{-2}x^{-2} + a_{-1}x^{-1} + a_0 + \cdots + a_k x^k \in Z_a[x^{\pm 1}]_{-\ell}^k$. Since the context is clear, we will write η in place of $\eta_{f,(-\ell,k)}$.
By definition,

$$\eta(\alpha) = \sum_{i=1}^{\ell} a_{-i}(-1)^i g(x)^i f(0)^{-i} + \sum_{i=0}^{k} a_i x^i.$$

It follows that

$$\begin{aligned} \alpha - \eta(\alpha) &= \sum_{i=1}^{\ell} a_{-i}x^{-i} - \sum_{i=1}^{\ell} a_{-i}(-1)^i g(x)^i f(0)^{-i} \\ &= \sum_{i=1}^{\ell} a_{-i}\Big(x^{-i} - (-1)^i g(x)^i f(0)^{-i}\Big) \end{aligned} \tag{2}$$

Observe that $f(x) = x \cdot g(x) + f(0) \implies x^{-1} - f(x)\big(x^{-1}f(0)^{-1}\big) = -g(x)f(0)^{-1}$. An easy induction shows that there is $\beta(x) \in \mathbb{Z}_a[x^{\pm 1}]$ such that $x^{-i} - f(x)\beta(x) = (-1)^i g(x)^i f(0)^{-i}$. That is, $x^{-i} - (-1)^i g(x)^i f(0)^{-i} = 0 \bmod f(x)$. It then follows from eq. (2) that $\alpha - \eta(\alpha) = 0 \bmod f(x)$.

Lemma 2. *The set $\mathbb{Z}_a[x^{\pm 1}]^k_{-\ell}$ is a complete set of coset representatives of $L_a = L/aL$ as long as* $\deg f = k + \ell + 1$.

Proof. Observe that for any two $\alpha, \beta \in \mathbb{Z}_a[x^{\pm 1}]^k_{-\ell}$, $\alpha \neq \beta \bmod f$. It now suffices to recall from [8] that $\eta_{f,(-\ell,k)} : \mathbb{Z}_a[x^{\pm 1}]^k_{-\ell} \to \mathbb{Z}_a[x^{\pm 1}]^{k+\ell}_0$ is a free module isomorphism and $\mathbb{Z}_a[x^{\pm 1}]^{k+\ell}_0$ is a complete set of coset representatives for R_a when $\deg f = k + \ell + 1$.

Corollary 1. *$\eta = \eta_{f,(-\ell,k)} : \mathbb{Z}_a[x^{\pm 1}]^k_{-\ell} \to \mathbb{Z}_a[x^{\pm 1}]^{k+\ell}_0$ induces the identity homomorphism on L_a. That is, the mapping $L_a \to L_a$ defined by $\alpha + f\mathfrak{L}_a \mapsto \eta(\alpha) + f\mathfrak{L}_a$ is the identity homomorphism.*

We now define a version of the Learning With Errors problem over the Laurent polynomial ring modulo the principal ideal generated by f and show that it is at least as hard as its PLWE counterpart.

Definition 2 (Decisional Laurent LWE). *For all $\kappa \in \mathbb{N}$, let $f(x) = f_\kappa(x)$ be a polynomial of degree $n = n(\kappa)$, and let $q = q(\kappa)$ be a prime integer such that $f(0)$ is invertible modulo q. Let $L = \mathbb{Z}[x^{\pm 1}]/f\mathbb{Z}[x^{\pm 1}]$, $L_q = L/qL$, and $\mathcal{X}$ denote a distribution over R. For non-negative integers ℓ, k such that* $\deg f = \ell + k + 1$, *take the coset representatives of L to be $\mathbb{Z}[x^{\pm 1}]^k_{-\ell}$. The decisional-LLWE problem $\mathsf{LLWE}^{(-\ell,k)}_{f,q,\chi}$ is, for any $m = poly(\kappa)$, to distinguish the sets*

$$\{(a_i, a_i \cdot S + e_i)\}_{i\in[m]} \text{ and } \{(a_i, u_i)\}_{i\in[m]}$$

where S is sampled from the distribution $\mathcal{X}$, the a_i are uniform in R_q, the error polynomials e_i are sampled from $\mathcal{X}$, and the ring elements u_i are uniformly random over R_q.

Theorem 1. *If one can solve the $\mathsf{LLWE}^{(-\ell,k)}_{f,q,\mathcal{X}}$ problem with the polynomial f such that $f(0) \in \mathbb{Z}_q$ is a unit, then one can solve the $\mathsf{PLWE}_{f,q,\chi}$ problem with the same polynomial f, $\chi = \eta(\mathcal{X})$, and $s = \eta(S)$.*

Proof. We will use the following pair of mappings from proposition 1 and [8], respectively:

$$\begin{aligned} &\gamma : R_q \to L_q \text{ defined by } \alpha + f\mathfrak{R}_a \mapsto \alpha + f\mathfrak{L}_a, \text{ and} \\ &\eta = \eta_{f,(-\ell,k)} : \mathbb{Z}_q[x^{\pm 1}]^k_{-\ell} \to \mathbb{Z}_q[x^{\pm 1}]^{k+\ell}_0 \end{aligned} \tag{3}$$

The latter mapping simply switches between sets of coset representatives for L_q. Recall that $A^{(q)}_{s,\chi}$ is the PLWE distribution and $\mathcal{L}^{(q)}_{S,\mathcal{X}}$ is the LLWE distribution. We can map the elements $(a, [a \cdot s + e]_q) \in R_q \times R_q$ to their isomorphic images under γ^{-1}
$\left(\gamma^{-1}(a), \left[\gamma^{-1}(a) \cdot \gamma^{-1}(s) + \gamma^{-1}(e)\right]_q\right) = (a, [a \cdot s + e]_q) \in L_q \times {L_q}^1$. We then use

[1] γ acts like the identity on coset representatives.

η^{-1} to switch from coset representatives in $\mathbb{Z}_t[x^{\pm 1}]_0^{k+\ell}$ to coset representatives in $\mathbb{Z}_q[x^{\pm 1}]_{-\ell}^{k}$. By corollary 1, we see that
$\eta^{-1}(a \cdot s + e) = \eta^{-1}(a)\eta^{-1}(s) + \eta^{-1}(e) = \eta^{-1}(a) \cdot S + \eta^{-1}(e)$. Clearly $\eta^{-1}(a)$ is uniform in L_q and, since $\chi = \eta(\mathcal{X})$, there is $e' \leftarrow \mathcal{X}$ such that $e' = \eta^{-1}(e)$. Consequently, $\left(\eta^{-1}(a), \left[\eta^{-1}(a) \cdot S + \eta^{-1}(e)\right]_q\right)$ is an LLWE instance according to the distribution $\mathcal{L}^{(q)}_{S,\mathcal{X}}$. So, if we can solve the Decisional-LLWE$_{d,q,\mathcal{X}}$ problem with $\chi = \eta(\mathcal{X})$ and $s = \eta(S)$, then we can solve the Decisional-PLWE$_{d,q,\chi}$ problem.

3.2 When is Laurent LWE hard?

If $f = \Phi_{2n}$, and χ is an appropriate gaussian error distribution over R with mean 0 for which $\mathsf{PLWE}_{f,q,\chi}$ is hard, then $\mathsf{LLWE}^{(-\ell,k)}_{f,q,\chi}$ is also hard. This comes down to the fact that, in this case, $\eta_f^{-1}(\chi) = \chi$. We elaborate below.

Proposition 3. *If $f(x) = x^n + 1$ and χ is a spherical discrete gaussian distribution over L with $\mu = 0$ and diagonal covariance matrix $\sigma^2 I$, then $\eta^{-1}_{f,(-\ell,k)}(\chi) = \chi$ for all integers $\ell, k \geq 0$ such that $n = k + \ell + 1$.*

Sketch of proof. First observe that for $f(x) = x^n + 1$, η_f is defined by the correspondences $x \mapsto x$ and $x^{-1} \mapsto -x^{n-1}$. This means, depending on the choice of ℓ, k, that η_f^{-1} can be viewed as a composition of negacyclic permutations of the coefficient vector of its argument. So, for $e \leftarrow \chi$, the set coefficients of $\eta_f^{-1}(e)$ and e are the same up to sign. The covariance matrix of χ being $\sigma^2 I$ implies that sampling $e \leftarrow \chi$ can be done coefficient-wise, sampling each coefficient from a gaussian distribution over $\mathbb{Z}$ with mean 0 and variance σ^2. Since $\eta_f^{-1}(e)$ and e have the same set of coefficients up to sign, the distribution from which each coefficient is selected remains unchanged. Consequently, $\eta_f^{-1}(\chi) = \chi$.

Theorem 2 (corollary of proposition 3). *If χ is a spherical discrete gaussian distribution over L with mean 0 and diagonal covariance matrix $\sigma^2 I$, then $\mathsf{LLWE}^{(-\ell,k)}_{f,q,\chi}$ is as hard $\mathsf{PLWE}_{f,q,\chi}$.*

We are unsure whether the LLWE and PLWE problems are actually equivalent, or whether there are polynomials f and distributions $\chi, \mathcal{X}$ with $\eta_f(\mathcal{X}) = \chi$ for which one of $\mathsf{PLWE}_{f,q,\chi}$ and $\mathsf{LLWE}^{(-\ell,k)}_{f,q,\mathcal{X}}$ is hard while the other is not. For now, we relegate this investigation to future work.

4 The new scheme: HERatio

4.1 Encoding rationals

Despite removing much of the machinery required by previous works to encode rationals for HE, HERatio does require some pre-processing – one must compute a bounded base-b expansion of a rational and then replace b by the unknown x to get a Laurent polynomial. We elaborate below on encode/decode and its correctness conditions.

$\mathsf{HERatio.Encode}\big(m, b, (-\ell, k)\big)$: For a message $m \in \mathbb{Q}$, compute the base-b expansion of m to obtain $\sum_{-\infty}^{\infty} a_i b^i$, where the $a_i \in \mathbb{Z}_b$. Truncate (if necessary) and replace everywhere b by x to obtain the Laurent polynomial $l(x) = \sum_{i=-\ell}^{k} a_i x^i \in \mathbb{Z}[x^{\pm 1}]_{-\ell}^{k}$ satisfying $l(b) \approx m$. Output $l(x)$.

$\mathsf{HERatio.Decode}\big(l'(x), b\big)$: Output $l'(b)$.

Correctness of decoding Let $r_1, \ldots, r_n \in \mathbb{Q}$ and $b \in \mathbb{Z}$ such that the base-b expansion of r_i is $l_i(b)$ for $l_i \in \mathbb{Z}[x^{\pm 1}]_{-\ell}^{k}$. Let $\mathcal{C}$ be an arithmetic circuit, and $l^* = \mathcal{C}(l_1, \ldots, l_n)$ be the evaluation of $\mathcal{C}$ at the l_i. Decoding is correct, i.e. $l^*(b) = \mathcal{C}\big(l_1(b), \ldots, l_n(b)\big)$, provided l^* remains in the *set* $\mathbb{Z}[x^{\pm 1}]_{-\ell}^{k}$.
An additional restriction must be imposed for correctness when the encoder is used with $\mathsf{HERatio}$. This is because the plaintext space is a ring of Laurent polynomials whose coefficients come from the ring $\mathbb{Z}_t$. The restriction is that computing the Laurent polynomial l^* above must not result coefficient overflow modulo t. This requires one to choose $b \leq t$ with b and t sufficiently far apart so that the desired circuits can be evaluated.

4.2 HERatio

Let $L = \mathbb{Z}[x^{\pm 1}]/\Phi_{2n}\mathbb{Z}[x^{\pm 1}]$, and for $a \in \mathbb{Z}$ let $L_a = L/aL$. The plaintext space is the ring L_t, for $t \geq 2$, and the ciphertext space is product ring $L_q \times L_q$ for $q \gg t$. The coset representatives of the elements of L are $\mathbb{Z}[x^{\pm 1}]_{-\ell}^{k}$, where ℓ, k are nonnegative integers satisfying $k + \ell + 1 = n$. We let λ be the security parameter and $\mathcal{X}$ be a discrete Gaussian distribution for which $\mathsf{LLWE}_{f,q,\mathcal{X}}^{(-\ell,k)}$ is hard (see theorem 2). $\mathsf{HERatio}$ is obtained from BFV by replacing everywhere R by L, R_t by L_t, and $R_q \times R_q$ by $L_q \times L_q$.

$\mathsf{HERatio.SecretKeyGen}$: Sample $s \in L$ with coefficients uniformly distributed in $\{-1, 0, 1\}$.
Output

$$\mathsf{sk} = s$$

$\mathsf{HERatio.PublicKeyGen}(\mathsf{sk})$: Let $s = \mathsf{sk}$. Sample $a \leftarrow L_q$, and $e \leftarrow \mathcal{X}$.
Output

$$\mathsf{pk} = \Big(\big[-\big(as + e\big)\big]_q, a\Big) \in L_q \times L_q$$

$\mathsf{HERatio.EvalKeyGen}(\mathsf{sk})$: For $i = 0, \ldots, \ell$, where $w \geq 2$ and $\ell = \lfloor \log_w q \rfloor$, sample $a_i \leftarrow L_q$, and $e_i \leftarrow \mathcal{X}$. Let

$$\mathsf{evk}[i] = \big([-(a_i s + e_i) + w^i s^2]_q, a_i\big) \in L_q \times L_q$$

Output the vector of pairs

$$\mathsf{evk} = \big(\mathsf{evk}[0], \ldots, \mathsf{evk}[\ell]\big)$$

$\mathsf{HERatio.Enc}(\mathsf{pk}, \ell \in L_t)$: Let $\Delta = \left\lfloor \frac{q}{t} \right\rfloor$ and $\mathsf{pk} = (p_0, p_1)$. Sample $u \in L$ with coefficients uniform in $\{-1, 0, 1\}$, and $e_0, e_1 \leftarrow \chi$.
Output

$$\mathsf{ct} = \big([\Delta\ell + p_0 u + e_0]_q, [p_1 u + e_1]_q\big) \in L_q \times L_q$$

$\mathsf{HERatio.Dec}(\mathsf{sk}, \mathsf{ct} \in L_q \times L_q)$: Let $s = \mathsf{sk}$ and $\mathsf{ct} = (c_0, c_1)$.
Output

$$\ell'(X) = \left[\left\lfloor \frac{t}{q} [c_0 + c_1 s]_q \right\rfloor \right]_t \in L_t$$

$\mathsf{HERatio.Add}(\mathsf{ct}_0, \mathsf{ct}_1)$:
Output

$$\big(ct_0[0] + ct_1[0], ct_0[1] + ct_1[1]\big) \in L_q \times L_q$$

$\mathsf{HERatio.PartialMult}(\mathsf{ct}_0, \mathsf{ct}_1)$: Denote $(c_0, c_1) = \mathsf{ct}_0$ and $(d_0, d_1) = \mathsf{ct}_1$.
Compute

$$c'_0 = \left[\left\lfloor \frac{t}{q} c_0 d_0 \right\rceil \right]_q,\ c'_1 = \left[\left\lfloor \frac{t}{q} (c_0 d_1 + c_1 d_0) \right\rceil \right]_q,\ c'_2 = \left[\left\lfloor \frac{t}{q} c_1 d_1 \right\rceil \right]_q,$$

Output

$$ct_{\text{prod}} = (c'_0, c'_1, c'_2) \in L_q \times L_q \times L_q$$

$\mathsf{HERatio.Relinearize}(\mathsf{ct}_{\text{prod}} \in L_q \times L_q \times L_q, \mathsf{evk})$: Denote $(c'_0, c'_1, c'_2) = ct_{\text{prod}}$.
Express c'_2 in base w, so that $c'_2 = \sum_{i=0}^{\ell} c_2'^{(i)} w^i$. Set

$$c_0 = c'_0 + \sum_{i=0}^{\ell} \mathsf{evk}[i][0] c_2'^{(i)},\ c_1 = c'_1 + \sum_{i=0}^{\ell} \mathsf{evk}[i][1] c_2'^{(i)},$$

Output

$$(c_0, c_1) \in L_q \times L_q$$

$\mathsf{HERatio.Mult}(ct_0, ct_1, \mathsf{evk})$:
Output

$$\mathsf{HERatio.Relinearize}\big(\mathsf{HERatio.PartialMult}(ct_0, ct_1), \mathsf{evk}\big)$$

4.3 Correctness of HERatio

In [17], Fan and Vercauteren analyze noise growth and derive correctness conditions for BFV using the polynomial infinity norm $\| \cdot \|_\infty$. Since this norm is defined coefficient-wise, switching from "classic" to Laurent polynomials does not change their analysis. This means that HERatio inherits the correctness conditions from [17] (in particular, lemma 1 and theorem 1).

5 Comparison with BFV

Since HERatio is obtained from BFV by simply swapping the ring of polynomials $\mathbb{Z}[x]/\Phi_{2n}\mathbb{Z}[x]$ for the ring of Laurent polynomials $\mathbb{Z}[x^{\pm 1}]/\Phi_{2n}\mathbb{Z}[x^{\pm 1}]$, we expect the performance of the two schemes to be very similar. With this in mind, we compare separately the encodings which allow HERatio and BFV, respectively, to work with rational numbers. The encoding used for BFV is that of [8] (see eq. (1)), and the encoding for HERatio is detailed in Section 4.1. We emphasize that the purpose of our implementations is to show that HERatio and BFV have similar performance when implemented the same way.

5.1 Implementation

We divided our results into 4 distinct groups, namely *Codec Operations* (encoding and decoding), *Key Generation*, *Ciphertext Generation*, and *Homomorphic Operations*, although some of them show dependent relationships. We measured the average time execution and memory used on each operation in order to analyze comparatively both HERatio and BFV schemes, along with their respective codecs HERatio.Encode, HERatio.Decode, and the rational encoder and decoder suggested in [8].

We chose messages $m_0 = 12345.678$ and $m_1 = 947.1273$, the additive scalar $s_a = 4$, and multiplicative scalar $s_m = 42.122$ to perform all calculations. The variables were initially defined either as 64-bits float numbers or 64-bits integers. However, during the execution these inputs were converted into arbitrary-size numbers to preserve correctness. For comparison purposes, we used the same parameters for HERatio and BFV, these are polynomials of length 32, with $q = 9,876,523,525, t = 2,131, \sigma = 3.19$, 10 as the expansion base, and $w = 128$ for the relinearization base for both BFV and HERatio.

The experiments were implemented with the Golang programming language version 1.19, on a MacBook Pro 15-inch, MacOS Monterey 12.7.1, 2.7GHz Quad-Core Intel(R) Core(TM) i7, 16GB 2133 MHz LPDDR3, 500GB SSD. Each operation was analyzed by evaluating the mean of $1,000$ executions, where the runtime and memory allocated were measured. Note that the implementation of BFV did not include any optimizations, though any existing optimization of BFV could be translated to an equivalent optimization of HERatio.

Codec Operations For Codec operations, we measured the performance of encoding a message m_0 as c_0 through both codecs, such that $c_0 = \mathsf{Encode}(m_0)$. Furthermore, we also executed the reverse operation to recover the original message from code c_0, such that $m_0' = \mathsf{Decode}(c_0)$.

The HERatio codec was 0.0567 ms faster for encoding a message, and 0.0025 ms faster for decoding.

Key Generation The key generation analysis measured the functions SecretKeyGen, PublicKeyGen, and EvalKeyGen, where all procedures used the same pseudo-random source. The $sk = \mathsf{SecretKeyGen}()$ function is equivalent for both schemes,

	Avg. Time (ms)			Memory (MB)	
Operation	[8]	HERatio		[8]	HERatio
Encoding	0.0619	0.0567	+8.40%	0.0227	0.0223
Decoding	0.0270	0.0245	+1.76%	0.0096	0.0083

Table 1: Codec Results.

whereas $pk = \mathsf{PublicKeyGen}(sk)$, and $evk = \mathsf{EvalKeyGen}(sk)$ differ in their internal polynomial multiplication.

	Avg. Time (ms)			Memory (MB)	
Operation	BFV	HERatio		BFV	HERatio
Secret Key	0.1066	0.1139	-6.40%	0.0324	0.0344
Public Key	5.9967	5.9625	+0.57%	3.2840	3.3453
Evaluation Key	56.2224	55.4312	+1.40%	29.2504	28.0354
Encryption	0.5886	0.5940	-0.90%	0.2867	0.2900
Decryption	0.3501	0.3402	+2.82%	0.1733	0.1756
Scalar Addition	0.0669	0.0719	-6.95%	0.0324	0.0344
Ciphertext Addition	0.1608	0.1598	+0.62%	0.0740	0.0767
Scalar Multiplication	0.1592	0.1551	+2.57%	0.0759	0.0744
Ciphertext Multiplication	3.1545	3.2326	-2.41%	1.6593	1.6857

Table 2: Operation Results.

HERatio and BFV had a differential of 0.0073 milliseconds (ms) for the secret key generation, a negligible difference that can be used as a threshold for comparison since the implementation of both functions are nearly equal. For public key generation HERatio was 0.0342 ms faster, and 0.7912 ms ahead for evaluation key.

Ciphertext Generation On cipher functions we had a small difference, where HERatio encrypted code c_0 as $ct_0 = \mathsf{Encrypt}(pk, c_0)$ 0.0054 ms slower than BFV, and decrypted the same ciphertext as $c_0' = \mathsf{Decrypt}(sk, ct_0)$ 0.0099 faster.

When we consider the difference between equivalent operations for both schemes, such as SecretKeyGen, we notice that the generation of ciphertexts have a negligible margin.

Homomorphic Operations We homomorphically tested additive and multiplicative operations (Table 2) between ciphertexts (i.e., ct_0 and ct_1) and additive (i.e., s_a) and multiplicative scalars (i.e., s_m).

HERatio executed the ciphertext addition and scalar multiplication, respectively, 0.001 ms and 0.004 ms faster than BFV, whereas the scalar addition and ciphertext multiplication were 0.005 ms and 0.0781 ms slower. The memory usage did not increase in a rate that was relevant for the experiment.

6 Conclusion and Future Work

We have presented a new variant of the LWE problem based on Laurent polynomials, and constructed a new variant of the BFV scheme based on this problem. The plaintext space of our scheme, HERatio, is a set of Laurent polynomials with exponents ranging from $-\ell$ to k. The exponent range can be chosen to suit the goal application, and does not affect the hardness of the new LWE variant. The main appeal of HERatio is that it can work with rationals via their bounded base-b expansions – all one must do is replace b by the unknown x. While enjoying efficiency comparable to BFV, HERatio is more mathematically streamlined than prior art, and also mitigates the difficulty in choosing parameters to make sure a rational encoding "plays well" with the underlying HE scheme.

Future Work As mentioned in Section 3.2 we are unsure of whether $\mathsf{LLWE}_{f,q,\mathcal{X}}^{(-\ell,k)}$ and $\mathsf{PLWE}_{f,q,\chi}$ are equivalent when $\eta_f(\mathcal{X}) = \chi$. It is our hope that a more detailed analysis of $\mathsf{LLWE}_{f,q,\mathcal{X}}^{(-\ell,k)}$ yields better insight into the Learning With Errors problems.

References

1. Martin R. Albrecht. On dual lattice attacks against small-secret LWE and parameter choices in HElib and SEAL. Cryptology ePrint Archive, Report 2017/047, 2017. `https://eprint.iacr.org/2017/047`.
2. Seiko Arita and Shota Nakasato. Fully homomorphic encryption for point numbers. Cryptology ePrint Archive, Report 2016/402, 2016. `https://ia.cr/2016/402`.
3. Charlotte Bonte, Carl Bootland, Joppe W. Bos, Wouter Castryck, Ilia Iliashenko, and Frederik Vercauteren. Faster homomorphic function evaluation using non-integral base encoding. In Wieland Fischer and Naofumi Homma, editors, *CHES 2017*, volume 10529 of *LNCS*, pages 579–600. Springer, Heidelberg, September 2017.
4. Joppe W. Bos, Kristin E. Lauter, and Michael Naehrig. Private predictive analysis on encrypted medical data. *Journal of biomedical informatics*, 50:234–43, 2014.
5. Zvika Brakerski. Fully homomorphic encryption without modulus switching from classical gapsvp. In *Annual Cryptology Conference*, pages 868–886. Springer, 2012.
6. Zvika Brakerski, Adeline Langlois, Chris Peikert, Oded Regev, and Damien Stehlé. Classical hardness of learning with errors. In Dan Boneh, Tim Roughgarden, and Joan Feigenbaum, editors, *45th ACM STOC*, pages 575–584. ACM Press, June 2013.

7. Zvika Brakerski and Vinod Vaikuntanathan. Fully homomorphic encryption from ring-LWE and security for key dependent messages. In Phillip Rogaway, editor, *CRYPTO 2011*, volume 6841 of *LNCS*, pages 505–524. Springer, Heidelberg, August 2011.
8. Wouter Castryck, Ilia Iliashenko, and Frederik Vercauteren. Homomorphic $\mathrm{SIM}^2\mathrm{D}$ operations: Single instruction much more data. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part I*, volume 10820 of *LNCS*, pages 338–359. Springer, Heidelberg, April / May 2018.
9. Hao Chen, Kim Laine, Rachel Player, and Yuhou Xia. High-precision arithmetic in homomorphic encryption. In *Cryptographers' Track at the RSA Conference*, pages 116–136. Springer, 2018.
10. Jung Hee Cheon, Jinhyuck Jeong, Joohee Lee, and Keewoo Lee. Privacy-preserving computations of predictive medical models with minimax approximation and non-adjacent form. In Michael Brenner, Kurt Rohloff, Joseph Bonneau, Andrew Miller, Peter Y. A. Ryan, Vanessa Teague, Andrea Bracciali, Massimiliano Sala, Federico Pintore, and Markus Jakobsson, editors, *FC 2017 Workshops*, volume 10323 of *LNCS*, pages 53–74. Springer, Heidelberg, April 2017.
11. Jung Hee Cheon, Andrey Kim, Miran Kim, and Yong Soo Song. Homomorphic encryption for arithmetic of approximate numbers. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT 2017, Part I*, volume 10624 of *LNCS*, pages 409–437. Springer, Heidelberg, December 2017.
12. A. Costache, N.P. Smart, S. Vivek, and A. Waller. Fixed point arithmetic in SHE scheme. Cryptology ePrint Archive, Report 2016/250, 2016. `https://eprint.iacr.org/2016/250`.
13. Ana Costache and Nigel P. Smart. Which ring based somewhat homomorphic encryption scheme is best? In Kazue Sako, editor, *CT-RSA 2016*, volume 9610 of *LNCS*, pages 325–340. Springer, Heidelberg, February / March 2016.
14. Anamaria Costache, Nigel P. Smart, and Srinivas Vivek. Faster homomorphic evaluation of discrete Fourier transforms. In Aggelos Kiayias, editor, *FC 2017*, volume 10322 of *LNCS*, pages 517–529. Springer, Heidelberg, April 2017.
15. Julien Devevey, Amin Sakzad, Damien Stehlé, and Ron Steinfeld. On the integer polynomial learning with errors problem. In Juan Garay, editor, *PKC 2021, Part I*, volume 12710 of *LNCS*, pages 184–214. Springer, Heidelberg, May 2021.
16. Nathan Dowlin, Ran Gilad-Bachrach, Kim Laine, Kristin Lauter, Michael Naehrig, and John Wernsing. Manual for using homomorphic encryption for bioinformatics. *Proceedings of the IEEE*, PP:1–16;, 02 2017.
17. Junfeng Fan and Frederik Vercauteren. Somewhat practical fully homomorphic encryption. *IACR Cryptol. ePrint Arch.*, 2012:144, 2012.
18. Luke Harmon, Gaetan Delavignette, Arnab Roy, and David Silva. Pie: p-adic encoding for high-precision arithmetic in homomorphic encryption. In Mehdi Tibouchi and XiaoFeng Wang, editors, *Applied Cryptography and Network Security*, pages 425–450, Cham, 2023. Springer Nature Switzerland.
19. Johan Håstad. Pseudo-random generators under uniform assumptions. In *22nd ACM STOC*, pages 395–404. ACM Press, May 1990.
20. J Hoffstein and JH Silverman. Optimizations for ntru. public-key cryptography and computational number theory, 2002.
21. Angela Jäschke and Frederik Armknecht. Accelerating homomorphic computations on rational numbers. In Mark Manulis, Ahmad-Reza Sadeghi, and Steve Schneider, editors, *ACNS 16*, volume 9696 of *LNCS*, pages 405–423. Springer, Heidelberg, June 2016.

22. Kristin E. Lauter, Adriana López-Alt, and Michael Naehrig. Private computation on encrypted genomic data. In Diego F. Aranha and Alfred Menezes, editors, *LATINCRYPT 2014*, volume 8895 of *LNCS*, pages 3–27. Springer, Heidelberg, September 2015.
23. Richard Lindner and Chris Peikert. Better key sizes (and attacks) for LWE-based encryption. In Aggelos Kiayias, editor, *CT-RSA 2011*, volume 6558 of *LNCS*, pages 319–339. Springer, Heidelberg, February 2011.
24. Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 1–23. Springer, Heidelberg, May / June 2010.
25. Michael Naehrig, Kristin Lauter, and Vinod Vaikuntanathan. Can homomorphic encryption be practical? In *Proceedings of the 3rd ACM Workshop on Cloud Computing Security Workshop*, CCSW '11, page 113–124, New York, NY, USA, 2011. Association for Computing Machinery.
26. Damien Stehlé, Ron Steinfeld, Keisuke Tanaka, and Keita Xagawa. Efficient public key encryption based on ideal lattices. In Mitsuru Matsui, editor, *ASIACRYPT 2009*, volume 5912 of *LNCS*, pages 617–635. Springer, Heidelberg, December 2009.