# Computing generator in cyclotomic integer rings

## A $L_{|\Delta_{\mathbb{K}}|}(1/2)$ algorithm for the Principal Ideal Problem and application to the cryptanalysis of a FHE scheme

Thomas Espitau[1], Pierre-Alain Fouque[2], Alexandre Gélin[1], and Paul Kirchner[3]

[1] Sorbonne Universités, UPMC Paris 6, UMR 7606, LIP6, Paris, France
thomas.espitau@lip6.fr, alexandre.gelin@lip6.fr
[2] Institut Universitaire de France, Paris, France and Université de Rennes 1, France
pierre-alain.fouque@univ-rennes1.fr
[3] École Normale Supérieure, Paris, France
paul.kirchner@ens.fr

**Abstract.** The Principal Ideal Problem (resp. Short Principal Ideal Problem), shorten as PIP (resp. SPIP), consists in finding a generator (resp. short generator) of a principal ideal in the ring of integers of a number field. Several lattice-based cryptosystems rely on the presumed hardness of these two problems. Yet, in practice, most of them do not use an arbitrary number field but a power-of-two cyclotomic field. The Smart and Vercauteren fully homomorphic encryption scheme and the multilinear map of Garg, Gentry and Halevi epitomize this common restriction. Recently, Cramer, Ducas, Peikert and Regev show that solving the SPIP in such cyclotomic rings boils down to solving the PIP. We complete their result by an algorithm that solves PIP in cyclotomic fields in subexponential time $L_{|\Delta_{\mathbb{K}}|}(1/2) = 2^{N^{1/2+o(1)}}$, where $\Delta_{\mathbb{K}}$ denotes the discriminant of the number field and $N$ its degree. This asymptotic complexity could be compared with the one obtained by Biasse and Fieker method, that aims at finding a generator as we do, but runs in $L_{|\Delta_{\mathbb{K}}|}(2/3)$. Besides this theoretical improvement, our algorithm allows to recover in practice the secret key of the Smart and Vercauteren scheme, for the smallest proposed parameters.[4]

## 1 Introduction

**Hard problem in Lattices.** Lattice-based problems appear to be among the most attractive alternatives to the integer factorization and discrete logarithm problems due to their conjectured resistance to quantum computations. Fortunately, all cryptographic primitives can be instantiated on the hardness of solving lattice problems, such as signature, basic encryption, Identity Based Encryption (IBE) as well as more powerful cryptosystems like Fully Homomorphic

[4] This work has been supported in part by the European Union's H2020 Programme under grant agreement number ICT-644209.

Encryption (FHE) [19]. All these schemes do not rely on the same lattice-based problem but on several. For instance, the NTRU cryptosystem [22], which is one of the most efficient encryption scheme related to our purpose, is based on the Shortest Vector Problem (SVP). Besides, the authors of NTRU were the first to consider specific kinds of lattices, namely related to polynomial rings. This idea enlightened another lattice-based problem that is now the most prolific in terms of schemes [31,28,40,29,30]: the Ring Learning With Error problem (Ring-LWE). Cryptosystems based on it present both an efficient key size reduction and improved performance (for instance decryption, encryption and signature are faster than with arbitrary lattices). Yet, Ring-LWE belongs to the specific family of *ideal-lattice* problems, which stem from algebraic number theory. This raises a potential drawback, since those lattices carry more structure than classical lattices, as they are derived from ideals in integer ring of number fields.

**SPIP and PIP.** Another presumably hard problem related to these ideals is called the Short Principal Ideal Problem (SPIP). It consists in finding a short[5] generator of an ideal, assuming it is principal. For instance, recovering the secret key from the public key in the Smart and Vercauteren Fully Homomorphic Encryption scheme [39] and in the Garg, Gentry and Halevi Multilinear Map scheme [18], consists in solving a SPIP. This problem turns out to hinge on two distinct phases: on the one hand finding an arbitrary generator – known as the Principal Ideal Problem (PIP)– and on the other hand reduce such a generator to a short one. The problem of finding a generator in a principal ideal, which is the aim of this article, or even testing the principality of an ideal, are difficult problems in algorithmic number theory, as precised in [14, Chapter 4] or [41, Section 7].

**From SPIP to PIP in Cyclotomic Fields.** Quite recently, Cramer, Ducas, Peikert and Regev demonstrate at Eurocrypt 2016 [16] how it is possible in polynomial-time to recover a small generator in prime-power cyclotomic fields, given an arbitrary generator. Before this paper, Campbell, Groves and Shepherd [11] first proposed an efficient algorithm for reduction, essentially by decoding the log-unit lattice with the LLL algorithm. However, they do not provide any rigorous proof of soundness, even though experiments confirm their claims, as well as implementations and tests performed in [36].

Studying SPIP and PIP in this very specific class of number fields is motivated by the concrete instantiations of the various schemes. Again the Smart and Vercauteren FHE scheme [39] and the Garg, Gentry and Halevi Multilinear Map scheme [18] epitomize this restriction to cyclotomic fields.

**From class group computations to PIP.** Based on earlier work of Hafner and McCurley [21] for quadratic number fields, Buchmann describes in [10]

[5] Short means that we have a norm. In our case, it is derived from the canonical embedding of the number field into a Euclidean space.

an algorithm for computing a finite presentation for the ideal class group and group of units for arbitrary-degree number fields $\mathbb{K}$. The first step consists in selecting a *factor base* $\mathcal{B}$ of all prime ideals of $\mathcal{O}_\mathbb{K}$ of norm smaller than a bound $B$, such that the classes of elements in $\mathcal{B}$ generate the class group [1,3]. The algorithm searches for relations among the elements of $\mathcal{B}$: given a random power-product $\mathcal{I}$ of ideals from the factor base, it computes a reduced representative $\mathcal{J}$ in the ideal class of $\mathcal{I}$ using lattice-reduction algorithms such as LLL or BKZ. If $\mathcal{J}$ splits on $\mathcal{B}$, a relation in the class group is found. Once sufficiently many relations are found, they entirely describe the class group and a Smith normal form yields the group structure. Buchmann algorithm has a running time exponential in the dimension: when taking into account the extension degree $N$, it is $\exp(O(N \log N)\sqrt{\log |\Delta_\mathbb{K}| \cdot \log\log |\Delta_\mathbb{K}|})$, where $\Delta_\mathbb{K}$ is the discriminant of the field. Its application to solving the PIP and mounting an attack is considered in [39]. Biasse and Fieker were the first to allow the dimension to tend to infinity and reach a subexponential complexity, in both the discriminant and the dimension.

**State-of-the-art.** A quantum polynomial-time algorithm for PIP is described by Biasse and Song at SODA 2016 [8] and a classical algorithm for any number field may be derived from the work of Biasse and Fieker presented at ANTS 2014 [6]. If we follow the lines of [6], we end up with a $2^{N^{2/3+o(1)}}$ subexponential algorithm that breaks the scheme exposed in [39], where $N$ denotes the dimension of the number field.

In a preprint [7] on arXiv, Biasse[6] gave an algorithm, close to ours, but different in the way the descent is implemented, with similar complexity. However, this article was withdrawn with a comment explaining that *This note (...) contains too many mistakes for public dissemination.* According to [7], since the approach uses heuristics, *it will be important to design an efficient implementation. To properly compare it with the existing methods, it will have to incorporate many classical practical improvements such as the use of a small factor base following the bounds described in [ 3 ], fast modular linear algebra to test the rank of the relation matrix, optimized methods to intersect an ideal with a subfield, (...). Besides the cryptographic applications, it will also be interesting to see how large the degree of the field can be pushed in the ideal class group computation. Indeed, the methods implemented in the main computer algebra softwares behave extremely badly when the degree of the number field exceeds 100...*

**Our results.** In this paper, we introduce a subexponential algorithm which solves the Principal Ideal Problem in power-of-two cyclotomic fields of dimension $N$ in $L_{|\Delta_\mathbb{K}|}(1/2) = 2^{N^{1/2+o(1)}}$ and provides a complete implementation. This new algorithm is designed in the same manner as discrete logarithm algorithms, namely involving relation collection and descent phase. We use in a crucial way

[6] The author updates a new version on September 30-th, where he obtains similar results to ours, using similar methods.

recent results on lattices having a sublattice with small determinant due to Cheon and Lee [13].

We are able to recover in practice a generator in the field $\mathbb{Q}(\zeta_{512})$. Such parameters are proposed by Smart and Vercauteren as toy parameters in [39]. The implementation is challenging since it requires to implement a version of the Gentry-Szydlo algorithm [20]. To our knowledge, this is the first implementation of this algorithm: even in [18] the authors concede that this part was not implemented in their attack against NTRU. In practice we prefer to use the version of Gentry-Szydlo described by Kirchner in [23]. We also implement many number field algorithms to perform the collection of relations and the descent.

**Organization of the paper.** In Section 2, we recall mathematical results for lattices and algebraic number theory that we use in the rest of the paper. Then, Section 3 presents the principal ideal problem (PIP) and the cryptosystem based on this problem such as the Smart-Vercauteren fully homomorphic encryption scheme. Next, we describe the different steps of the algorithm to solve PIP in Section 4. Finally, Section 5 gives information about our experimentations.

## 2 Mathematical background

We recall briefly here basic facts on lattices and algebraic number theory. A more detailed introduction is provided in the Appendix.

**General notations.** For dealing with complexities, we introduce the $L$-notation, that is classical when presenting index calculus algorithms with subexponential complexity. Given two constants $a$ and $c$ with $a \in [0, 1]$ and $c \geq 0$, we denote by:

$$L_{|\Delta_{\mathbb{K}}|}(a, c) = e^{(c+o(1))(\log |\Delta_{\mathbb{K}}|)^a (\log\log |\Delta_{\mathbb{K}}|)^{1-a}},$$

where $o(1)$ tends to 0 as $|\Delta_{\mathbb{K}}|$, the discriminant of the number field tends to infinity. We also encounter the notation $L_{|\Delta_{\mathbb{K}}|}(a)$ when specifying $c$ is superfluous, that is considering quantities in $L_{|\Delta_{\mathbb{K}}|}(a, \mathcal{O}(1))$.

### 2.1 Lattices

Lattices are defined as additive discrete subgroups of $\mathbb{R}^n$, i.e. the integer span $L(\mathbf{b}_1, \ldots, \mathbf{b}_d) = \bigoplus_{i=1}^{d} \mathbb{Z}\mathbf{b}_i$ of a linearly independent family of vectors $\mathbf{b}_1, \ldots, \mathbf{b}_d$ in $\mathbb{R}^n$. Such a family is called a *basis* of the lattice, and is not unique. Nevertheless, all the bases of a given lattice have the same number of elements, $d$, which is called the *dimension* of the lattice. Among the infinite number of different bases of a $n$-dimensional lattice with $n \geq 2$, some have interesting properties, such as having reasonably small vectors and low orthogonality defect — that means that they are almost orthogonal.

The problem of finding such good bases is the aim of *lattice reduction*. There are in short two kinds of reduction algorithms: *approximation* algorithms on the one hand, like the celebrated LLL algorithm and its blockwise variants

such as BKZ and DBKZ [32], and *exact* algorithms on the other hand, such as enumeration or sieving, that are exponential in time and space. In high dimension, only approximation algorithms - which run in polynomial time in the dimension - can be used to find relatively short vectors, but usually not shortest ones.

**The DBKZ algorithm and Cheon's determinant trick.** In this part, we recall the complexity of DBKZ algorithm, introduced by Micciancio and Walter in [32], its approximation factor and a trick due to Cheon and Lee [13] that improves this factor for integer lattices with small determinant.

**Theorem 1 (Bounds for DBKZ output).** *The smallest vector output by DBKZ algorithm with block-size $\beta$ has a norm bounded by:*

$$\beta^{\frac{n-1}{2(\beta-1)}} \cdot \mathrm{Vol}(\mathcal{L})^{\frac{1}{n}}.$$

*The algorithm runs in time Poly$(n, size(\mathbf{B})) \cdot (3/2 + o(1))^{\beta/2}$, where $\mathbf{B}$ is the input basis and $(3/2 + o(1))^{\beta/2}$ stands for the cost of solving the Shortest Vector Problem in dimension $\beta$, using sieving techniques (see [2]).*

*Proof.* This is a direct application of [32, Theorem 1], where the Hermite constant $\gamma_\beta$ is upper bounded by $\beta$.

In a note [13] of 2015, Cheon and Lee suggest to convert the basis of an integer lattice having small determinant, to its Hermite normal form (HNF) before reducing it, for instance with the DBKZ algorithm. This algorithm seems to be folklore but this note gives a detailled analysis. We develop here this idea and derive corresponding bounds. For completeness purpose, the definition of HNF is recalled in Appendix A.1. More precisely we have:

**Lemma 1.** *Given $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_n]$ a basis in HNF of a $n$-dimensional lattice $\mathcal{L}$, we have for any $1 \leq i < n$:*

$$\mathrm{Vol}([\mathbf{b}_1, \ldots, \mathbf{b}_i]) \leq \mathrm{Vol}([\mathbf{b}_1, \ldots, \mathbf{b}_{i+1}]).$$

*In particular, for any sublattice $\mathcal{L}'$ generated by the $m$ first vectors of $\mathbf{B}$, we have $\mathrm{Vol}(\mathcal{L}') \leq \mathrm{Vol}(\mathcal{L})$.*

Remark that both the $n$-th root of the determinant and an exponential factor of $n$ appear in the bound of Theorem 1. Hence we can perform the DBKZ reduction on a sublattice only generated by the first $m$ columns of the HNF in order to minimize this upper bound, as a trade-off between these quantities. Explicitly we fix $m = \left\lfloor \sqrt{\frac{2\beta}{\log \beta} \log(\mathrm{Vol}(\mathcal{L}))} \right\rceil$ and run the algorithm of Figure 1 on the basis $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$:

1. Compute the HNF $(\mathbf{b'}_1, \ldots, \mathbf{b'}_n)$ of $\mathbf{B}$.
2. Run DBKZ with block-size $\beta$ on $(\mathbf{b'}_1, \ldots, \mathbf{b'}_m)$ with $m = \left\lfloor \sqrt{\frac{2\beta}{\log \beta} \log(\mathrm{Vol}(\mathcal{L}))} \right\rceil$
3. Return the first vector of the output of DBKZ.

Fig. 1: Approx-SVP algorithm with HNF+DBKZ with block-size $\beta$

**Theorem 2.** *For any $n$-dimensional integer lattice $\mathcal{L}$ such that $\mathrm{Vol}(\mathcal{L}) \leq \beta^{\frac{n^2}{2\beta}}$, the output $\mathbf{v}$ of the previous Approx-SVP algorithm satisfies:*

$$\|\mathbf{v}\| \leq \beta^{(1+o(1))\sqrt{2\log_\beta(\mathrm{Vol}(\mathcal{L}))/\beta}}.$$

*This algorithm takes time Poly$(n, size(\mathbf{B}))(3/2 + o(1))^{\beta/2}$.*

*Proof.* The condition on the covolume of $\mathcal{L}$ ensures that $m \leq n$.

Then, by Theorem 1 and Lemma 1 we have:

$$\begin{aligned}\|\mathbf{v}\| &\leq \beta^{\frac{m}{2\beta}} \cdot \mathrm{Vol}(\mathcal{L}')^{\frac{1}{m}} \\ &\leq \beta^{\frac{m}{2\beta}} \cdot \mathrm{Vol}(\mathcal{L})^{\frac{1}{m}} \\ &= \beta^{\sqrt{2\log_\beta(\mathrm{Vol}(\mathcal{L}))/\beta}}\end{aligned}$$

yielding the announced result.

### 2.2 Number Fields

Let $\mathbb{K} = \mathbb{Q}(\alpha)$ be a number field of degree $N$, then there exists a monic irreducible degree-$N$ polynomial $P \in \mathbb{Z}[X]$ such that $\mathbb{K} \simeq \mathbb{Q}[X]/(P)$. Denoting by $(\alpha_1, \ldots, \alpha_N) \in \mathbb{C}^N$ its distinct complex roots, each embedding (field homomorphism) $\sigma_i : \mathbb{K} \to \mathbb{C}$ is the evaluation of $\mathbf{a} \in \mathbb{K}$, viewed as a polynomial $\mod P$, at the root $\alpha_i$, i.e. $\sigma_i : \mathbf{a} \mapsto \mathbf{a}(\alpha_i)$. If we have $r_1$ real roots and $r_2$ pairs of complex roots ($N = r_1 + 2r_2$), we have $\mathbb{K} \otimes \mathbb{R} \simeq \mathbb{R}^{r_1} \times \mathbb{C}^{r_2}$ so that we can define a norm $\|\cdot\|$ over $\mathbb{K}$ as the canonical euclidean norm of $\mathbb{R}^{r_1} \times \mathbb{C}^{r_2}$ where the canonical embedding is defined as: $\sigma(\mathbf{x}) = (\sigma_1(\mathbf{x}), \ldots, \sigma_{r_1+r_2}(\mathbf{x})) \in \mathbb{R}^{r_1} \times \mathbb{C}^{r_2}$, where $\sigma_1, \ldots, \sigma_{r_1}$ are the real embeddings and $\sigma_{r_1+1}, \ldots, \sigma_N$ are the complex embeddings where $\sigma_{r_1+j}$ is paired with its complex conjugate $\sigma_{r_1+r_2+j}$. The number field $\mathbb{K}$ is viewed as an euclidean $\mathbb{Q}$-vector space endowed with the inner product $\langle \mathbf{a}, \mathbf{b} \rangle = \sum_\sigma \sigma(\mathbf{a})\bar{\sigma}(\mathbf{b})$ where $\sigma$ ranges over all the $r_1 + 2r_2$ embeddings $\mathbb{K} \to \mathbb{C}$. This defines the euclidean norm denoted $\|\cdot\|$. A simple equality between the field norm and the embeddings exists:

$$\mathcal{N}_{\mathbb{K}/\mathbb{Q}}(\mathbf{v}) = \prod_{i=1}^{N} \sigma_i(\mathbf{v}).$$

**Coefficient embedding and ideal lattices.** Let $\alpha$ be one of the roots $\alpha_i$ (it may differs from the initial $\alpha$ is this one is not an algebraic integer). Considering

the natural isomorphism between $\mathbb{Z}[\alpha] \subset \mathcal{O}_{\mathbb{K}}$ and $\mathbb{Z}[X]/(P)$ gives rise to an embedding of $\mathbb{Z}[\alpha]$ trough the coefficients of associated polynomials. More precisely, we have the following sequence of abelian groups

$$\begin{array}{ccccc} \mathbb{Z}^N & \xhookrightarrow{\iota} & \mathbb{Z}[X] & \xrightarrow{\pi} & \mathbb{Z}[X]/(P) \simeq \mathbb{Z}[\alpha] \\ (c_0, \cdots, c_{N-1}) & \longmapsto & \sum_{0 \leq i < N} c_i X^i & \longmapsto & \sum_{0 \leq i < N} c_i \alpha^i, \end{array}$$

defining the announced embedding by coefficients as $\mathcal{C} = \iota^{-1} \circ \pi^{-1}$. Such an embedding provides a norm in the field, namely: $\|\mathbf{a}\|_{\mathcal{C}} = \|\mathcal{C}(\mathbf{a})\|_2$.

Let state a basic result on the link between field norm and polynomial representation:

**Lemma 2.** *For algebraic integers defined as polynomials in $\alpha$, namely $\mathbf{a} = T(\alpha)$ for $T \in \mathbb{Z}[X]$, we can bound the norm by*

$$|\mathcal{N}_{\mathbb{K}/\mathbb{Q}}(\mathbf{a})| \leq (N+1)^{m/2} (m+1)^{N/2} H(T)^N H(P)^m,$$

*where $m = \deg T$, $N = \deg P$ and $H(P)$ is the absolute maximum of the coefficients of $P$.*

*Proof.* Remark first that the norm of this element corresponds to the resultant of the polynomials $T$ and $P$ [14, Proposition 4.3.4]. Then we apply the bounds of [9, Theorem 7] for the resultant of two polynomials and conclude.

As a result, we can directly relate the norm of the embedding with the field norm:

**Corollary 1.** *For any $\mathbf{a} \in \mathbb{Z}[\alpha]$: $|\mathcal{N}_{\mathbb{K}/\mathbb{Q}}(\mathbf{a})|^{\frac{1}{N}} \leq (N+1) \cdot H(P) \cdot \|\mathbf{a}\|_{\mathcal{C}}$.*

**Canonical embedding and ideals.** A remarkable property of the canonical embedding is the way it represents the ring of integers and more generally every integral ideal. Indeed, the embedding $\sigma(\mathfrak{a})$ of any integral ideal $\mathfrak{a}$ is a Euclidean lattice. In particular, for the ring of integers, we have that $\sigma(\mathcal{O}_{\mathbb{K}})$ is a lattice. Its (co)volume is called the *discriminant* $\Delta_{\mathbb{K}}$ of the field $\mathbb{K}$. Therefore, one can compute the discriminant as a determinant: for $(\mathbf{b}_1, \ldots, \mathbf{b}_N)$ an integral basis of $\mathcal{O}_{\mathbb{K}}$, we have

$$\Delta_{\mathbb{K}} = \left( \det \begin{pmatrix} \sigma_1(\mathbf{b}_1) & \sigma_1(\mathbf{b}_2) & \cdots & \sigma_1(\mathbf{b}_N) \\ \sigma_2(\mathbf{b}_1) & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ \sigma_N(\mathbf{b}_1) & \cdots & \cdots & \sigma_N(\mathbf{b}_N) \end{pmatrix} \right)^2.$$

Loosely speaking, the discriminant is a size measure of the integer ring. That is why we use it to express the complexity when we work with number fields or rings of integers. Moreover, it acts as a proportionality coefficient between the norm of an ideal and the covolume of its embedding:

**Lemma 3.** *For any integral ideal $\mathfrak{a}$ of $\mathbb{K}$, we have: $\sigma(\mathfrak{a})$ is a lattice of $\mathbb{R}^N$ and*

$$\mathrm{Vol}(\sigma(\mathfrak{a})) = \sqrt{|\Delta_{\mathbb{K}}|} \mathcal{N}(\mathfrak{a}).$$

**Smoothness of ideals.** To evaluate the probability of smoothness of ideals, we need to refer to an unproven heuristic, directly derived from what has been proved for integers by Canfield, Erdős and Pomerance [12]. Let $\mathcal{P}(x, y)$ be the probability that a principal ideal of $\mathcal{O}_{\mathbb{K}}$ of norm bounded by $x$ is a power-product of prime ideals of norm bounded by $y$. Then, we have:

**Heuristic 1** *We assume that under the Generalized Riemann Hypothesis (GRH), the probability $\mathcal{P}(x, y)$ satisfies*

$$\mathcal{P}(x, y) \geq e^{-u \log u(1+o(1))} \quad \textit{for } u = \frac{\log x}{\log y}.$$

In the number field setting, the previous heuristic admits a neat rewriting in terms of the handy $L$-notation:

**Corollary 2.** *Let $x = \lfloor \log L_{|\Delta_{\mathbb{K}}|}(a, c) \rfloor$ and $y = \lceil \log L_{|\Delta_{\mathbb{K}}|}(b, c') \rceil$. Then assuming Heuristic 1, the probability $\mathcal{P}(x, y)$ that an ideal of $\mathcal{O}_{\mathbb{K}}$ of norm bounded by $x$ is a power-product of prime ideals of norm bounded by $y$ satisfies*

$$\mathcal{P}(x, y) \geq L_{|\Delta_{\mathbb{K}}|}\left(a - b, \frac{-c}{c'}(a - b)\right).$$

A similar assertion for smoothness of ideals was proved by Seysen [37] in 1985 for the quadratic case, but for arbitrary degree, it remains conjectural, even under GRH. This is one of the reasons why the complexity of the number field sieve (NFS) [26] is still a heuristic estimation.

### 2.3 Cyclotomic fields and Cyclotomic Integers

We denote by $\Phi_m$ the *$m$-th cyclotomic polynomial*, that is the unique irreducible polynomial in $\mathbb{Q}[X]$ dividing $X^m - 1$ that is not a divisor of any of the $X^k - 1$ for $k < m$. Its roots are thus the $m$-th primitive roots of the unity. Therefore, cyclotomic polynomials can be written in closed form as:

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

The $m$-th cyclotomic field $\mathbb{Q}(\zeta_m)$ is obtained by adjoining a primitive $m$-th root $\zeta_m$ of unity to the rational numbers. As such, $\mathbb{Q}(\zeta_m)$ is isomorphic to the splitting field $\mathbb{Q}[X]/(\Phi_m)$. Its degree over $\mathbb{Q}$ is $\deg(\Phi_m)$, that is $\varphi(m)$, where $\varphi$ is the Euler totient function. In this specific class of number fields, the ring of integer is precisely $\mathbb{Z}[X]/(\Phi_m) \cong \mathbb{Z}[\zeta_m]$ (see [42], Theorem 2.6 for a proof of this statement).

The canonical embedding can also be easily presented since the embeddings are the linear functions sending $\zeta_m$ to $\zeta_m^j$, for $j \in (\mathbb{Z}/m\mathbb{Z})^*$. Since the roots come

by conjugate pairs ($\zeta_m^j = -\zeta_m^{m-j}$ for all $j$) we can write down the Log-embedding by indexing over the quotient $G = (\mathbb{Z}/m\mathbb{Z})^* / \{-1, 1\}$:

$$\begin{aligned} \mathrm{Log}(x) : \quad \mathbb{K} \quad &\longrightarrow \mathbb{R}^{\varphi(m)/2} \\ P \mod \Phi_m &\mapsto \left(\log |P(\zeta_m^j)|\right)_{j \in G}. \end{aligned}$$

Consequently, the discriminant has a closed form expression [42, Proposition 2.7]:

$$\Delta_{\mathbb{Q}(\zeta_m)} = (-1)^{\varphi(m)/2} \frac{m^{\varphi(m)}}{\prod_{p|m} p^{\varphi(m)/(p-1)}},$$

where the product in the denominator is over primes $p$ dividing $m$.

*Example 1.* For a prime-power cyclotomic field, simplifications operate as

$$\left|\Delta_{\mathbb{Q}(\zeta_{p^k})}\right| = p^{(kp-k-1)p^{k-1}}.$$

In particular, when $p = 2$, $\left|\Delta_{\mathbb{Q}(\zeta_{2^{n+1}})}\right| = 2^{n2^n}$.

For power-of-two cyclotomic fields, we then have the following relation: $L_{|\Delta_{\mathbb{K}}|}(\alpha) = 2^{\mathcal{O}(N^\alpha \log(N))}$. Thus explicit complexity as $L_{|\Delta_{\mathbb{K}}|}(\alpha)$ or $2^{\mathcal{O}(N^\alpha \log(N))}$ is equivalent. We choose to use the $L$-notation, since it eases the exposition of the complexities presented in this paper.

### 2.4 Cyclotomic units

Giving the complete description of the units of a generic number field is a computationally hard problem of algorithmic number theory. However it is possible to describe a subgroup of finite index[7] of the unit group, called the *cyclotomic units*. This subgroup contains all the units that are products of numbers[8] of the form $\zeta_m^i - 1$ for any $1 \leq i \leq m$. More precisely we have:

**Lemma 4 (Lemma 8.1 of [42]).** *Let $m$ be a prime power, then the group $C$ of cyclotomic units is generated by $\pm\zeta_m$ and $(\mathbf{b}_i)_{1 \leq i \leq m}$, where*

$$\mathbf{b}_i = \frac{\zeta_m^i - 1}{\zeta_m - 1}.$$

The index of the subgroup of cyclotomic units in the group of units is $h^+(m)$, the class number of the totally real subfield of $\mathbb{Q}(\zeta_m)$ (see for instance [42]). In the case of power-of-two $m$, a well supported conjecture clarifies the value of $h^+$.

**Heuristic 2 (Weber's class number problem)** *We assume that for power-of-two cyclotomic fields, the class number of its totally real subfield is 1.*

Thus under Weber's heuristic, the cyclotomic units and the units coincide in the power-of-two cyclotomic fields.

[7] The *index* of $H$ as subgroup of $G$ is defined as the number of cosets of $H$ in $G$, which is also the cardinal of the quotient $G/H$.

[8] One should notice that if $m$ is a prime-power, $\zeta_m^i - 1$ is not a unit, but $\mathbf{b}_i$ is.

## 3 Principal-Ideal Problem and Cryptography

Among all the Fully Homomorphic Encryption (FHE) schemes proposed in the last decade, the security of a couple of them directly relies on the ability to find relatively short generators in principal ideals. This is the case of the proposal of Smart and Vercauteren [39], which is a simplified version of the original scheme of Gentry [19], or the similar Soliloquy scheme of Campbell, Groves and Shepherd [11] and even candidates for multilinear maps [18,25]. More formally, the underlying – presumably hard – problem is the following one, already known as SPIP (Short Principal Ideal Problem) or SG-PIP (Short Generator-Principal Ideal Problem): given some $\mathbb{Z}$-basis of a principal ideal with a promise that it possesses a "short" generator $\mathbf{g}$ for the Euclidean norm, find this generator or at least a short enough generator of this ideal.

The strategy to address this problem roughly splits in two main steps:

1. Given the $\mathbb{Z}$-basis of the ideal, find a generator, not necessarily short, that is $\mathbf{g} \cdot \mathbf{u}$ for a unit $\mathbf{u}$.
2. From this freshly discovered generator, find a short one.

Very recently, many results have allowed to deal with the second step. Indeed, Campbell, Groves and Shepherd [11] claimed in 2014 an – although unproven – efficient solution for power-of-two cyclotomic fields, confirmed by experiments conducted by Schank [36] in 2015. Eventually, the proof was provided by Cramer, Ducas, Peikert and Regev [16] in 2015. Throughout this paper, we focus on the resolution of the first step, known as PIP (Principal Ideal Problem). Nonetheless for completeness we present briefly the reduction from SPIP to PIP in section 4.4.

As a direct illustration of the resolution of this problem, we present an attack on the scheme that Smart and Vercauteren present in [39], which leads to a *full key recovery*. This attack is our key thread through the exposition of the algorithm. Before going any further in the details of the attack, we recall in Figure 2 the key generation process in the case of cyclotomic field of power-of-two index. This instantiation is the one chosen by the authors for presenting their implementation results.

---

1. Fix the security parameter $N = 2^n$.
2. Let $F(X) = X^N + 1$ be the polynomial defining the cyclotomic field $\mathbb{K} = \mathbb{Q}(\zeta_{2N})$.
3. Let $G(X) = 1 + 2 \cdot S(X)$ for $S(X)$ of degree $N - 1$ with coefficients absolutely bounded by $2^{\sqrt{N}}$, such that the norm $\mathcal{N}(\langle G(\zeta_{2N})\rangle)$ is prime.
4. Let $\mathbf{g} = G(\zeta_{2N}) \in \mathcal{O}_{\mathbb{K}}$.
5. Return $(\mathsf{sk} = \mathbf{g}, \mathsf{pk} = \mathrm{HNF}(\langle \mathbf{g} \rangle))$

---

Fig. 2: Key Generation of the scheme [39].

*Remark 1.* The public key can be any $\mathbb{Z}$-basis of the ideal generated by $\mathbf{g}$, for instance any two-elements representation of this ideal. Precisely, [39] provides

the public key as a pair of elements that generates the lattice. This is always possible, see [14, Section 4.7.2]. We make the choice of the Hermite Normal Form representation[9].

As our attack consists in a full secret key recovery, realized directly from the public key, we do not mention here the encryption and decryption procedures. Even though this work tackles more on the principal ideal problem than on this reduction, we emphasize the fact that the output of this reduction to a short generator can be any one of the $\mathbf{g} \cdot \zeta_{2N}^i$, having same Euclidean norm for any $1 \leq i \leq 2N$. Nonetheless, this does not represent an issue, since all of these keys are equivalent with regard to the decryption procedure. In addition, in this precise construction of Smart and Vercauteren, the only odd coefficient of $G(X)$ is the last one, so that we may recover the exact generator $\mathbf{g}$ readily.

The whole complexity of our attack is subexponential, in $L_{|\Delta_{\mathbb{K}}|}(1/2) = 2^{N^{1/2+o(1)}}$. This beats the previous state-of-the-art in $L_{|\Delta_{\mathbb{K}}|}(2/3) = 2^{N^{2/3+o(1)}}$, derived from the combined work of [6] and [16].

## 4 Solving the PIP or how to perform a full key recovery?

We recall that our ultimate goal is to perform a full key recovery given only the public elements. As mentioned in [39], this problem is obviously much more difficult than recovering a plain-text from a cipher-text which is based on the bounded distance decoding problem and the security level is set according to this latter problem. We first give an overview of the whole strategy and then get an in-depth view of each part. But before going any further in the details of the attack, let us fix the notations and recurrent objects we are going to use. The number field where this story takes place is $\mathbb{Q}(\zeta_{2N})$, for $N = 2^n$, defined by the polynomial $X^N + 1$, in the same fashion as in Section 2.3. For the sake of notation simplicity, $\zeta_{2N}$ is simply denoted by $\zeta$. Our starting point is the public key, that is, a somewhat "bad" basis of the principal ideal $\mathcal{I} = \langle \mathbf{g} \rangle$, generated by the secret key $\mathbf{g}$.

Before any other operations, the dimension of the ideals involved is shrunk by half by reducing the problem to an equivalent one in the *totally real subfield* $\mathbb{Q}(\zeta + \zeta^{-1})$. This part of the algorithm is a straightforward consequence of the Gentry-Szydlo algorithm introduced in [20]. The problem is now reduced to the research of a generator of an ideal $\mathcal{I}^+$ in the totally real subfield. Then, the strategy appears to be recursive reductions of ideals, until we eventually reach a $B$-smooth ideal $\mathcal{I}^s$, for a fixed bound $B > 0$ and an algebraic integer $\mathbf{h}$ such that $\langle \mathbf{h} \rangle = \mathcal{I}^+ \cdot \mathcal{I}^s$. This is the *descent phase*.

The main ideal we are interested in now is $\mathcal{I}^s$. A generator of it has to be found and we use techniques similar to *class group computation*, namely by collecting relations between prime ideals of norm below $B$. Once sufficiently many pieces of information on the ideal $\mathcal{I}^s$ are obtained as these relations, performing

[9] The definition of the HNF is recalled for completeness in Appendix A.

linear algebra leads to the discovery of one of its generator $\mathbf{h_0}$. It permits then to derive the generator of the ideal $\mathcal{I}^+$: $\mathbf{h} \cdot \mathbf{h_0^{-1}}$. A generator of the public-key ideal is then obtained by lifting it from the totally real subfield to the initial number field $\mathbb{Q}(\zeta)$. It suffices to multiply the current generator by another integer obtained during the computation. Now PIP is solved, it only remains a final step to recover the secret key: perform the reduction from this generator to a short one, using the method by Cramer, Ducas, Peikert and Regev of [16].

Consequently, the full algorithm can be split in four main steps, which are, in a nutshell:

1. Perform a reduction from the cyclotomic field to its totally real subfield, allowing to work in smaller dimension.
2. Then a descent makes the size of involved ideals decrease.
3. Collect relations and run linear algebra to construct small ideals and a generator.
4. Eventually run the derivation of the small generator from a bigger one.

Let us now get into the details of all these parts.

### 4.1 Step 1: Reduction to the totally real subfield

Starting with the public key, we get a $\mathbb{Z}$-basis $(\mathbf{b}_1, \ldots, \mathbf{b}_N)$ of an ideal $\mathcal{I}$ belonging to the cyclotomic field $\mathbb{Q}(\zeta)$ of dimension[10] $N$. Undoubtedly, a curse of dimensionality occurs: the larger the dimension is, the harder it is to handle and even only represent such objects. However, salvation with regards to tractability comes here from the possibility to halve the dimension. The main part of this step relies on the so-called *Gentry-Szydlo* (GS) algorithm, first described in [20] as an attack on the NTRU scheme and later revised and generalized by Lenstra and Silverberg in [27].

This original algorithm takes as input a $\mathbb{Z}$-basis of an ideal $\mathcal{I}$ in the ring $\mathbb{Z}[X]/(X^N - 1)$ – with the promise to be principal – and the algebraic integer $\mathbf{u} \cdot \bar{\mathbf{u}}$, for $\mathbf{u}$ a generator of $\mathcal{I}$. Here, $\bar{\mathbf{u}}$ denotes the conjugate of $\mathbf{u}$ for the automorphism defined by $\zeta \mapsto \zeta^{-1}$. It then recovers in polynomial time the element $\mathbf{u}$. In our case, we can not perform the recovery of the generator $\mathbf{g}$, secret key of the scheme, since *a priori* we do not have access to any kind of information about the product $\mathbf{g} \cdot \bar{\mathbf{g}}$.

To overcome this difficulty, we introduce another integer $\mathbf{u} = \mathcal{N}(\mathbf{g})\, \mathbf{g}\, \bar{\mathbf{g}}^{-1}$, a trick already described by Garg, Gentry and Halevi in [18, Section 7.8.1]. One should notice that the norm factor is only there to avoid introduction of denominators in the definition of $\mathbf{u}$. Although $\mathbf{u}$ is still unknown at this point, thanks to the $\mathbb{Z}$-basis of $\langle \mathbf{g} \rangle$ we can construct a $\mathbb{Z}$-basis of $(\mathbf{u})$ and deriving the product $\mathbf{u} \cdot \bar{\mathbf{u}}$ which simply corresponds to $\mathcal{N}(\mathbf{g})^2$.

[10] The smallest security parameters of the Smart and Vercauteren scheme is $N = 256$

Hence, we get access to $\mathbf{u}$ in polynomial time using GS. From this freshly obtained element $\mathbf{u}$ we directly reconstruct $\mathbf{g}\,\bar{\mathbf{g}}^{-1}$ and using the basis of $\mathcal{I}$, we then introduce the family of vectors

$$\mathbf{c}_i = \mathbf{b}_i \left(1 + \frac{\bar{\mathbf{g}}}{\mathbf{g}}\right),$$

providing a basis of the ideal $\mathcal{I}^+$ generated by $\mathbf{g} + \bar{\mathbf{g}}$. The reader should notice that this ideal belongs to the totally real subfield $\mathbb{Q}(\zeta + \zeta^{-1})$, of index 2 in $\mathbb{Q}(\zeta)$. From now on, we denote by $\mathcal{O}_{\mathbb{K}}^+$ the ring of integers of $\mathbb{Q}(\zeta + \zeta^{-1})$, corresponding to $\mathcal{O}_{\mathbb{K}} \cap \mathbb{Q}(\zeta + \zeta^{-1})$.

Let suppose briefly that we know the generator $\mathbf{g} + \bar{\mathbf{g}}$ of $\mathcal{I}^+$. Then it would be sufficient to multiply it by $\frac{1}{1+\mathbf{g}\,\bar{\mathbf{g}}^{-1}}$ to recover the secret key $\mathbf{g}$. Hence, we have reduced the problem of finding a generator of the idea $\mathcal{I}$ belonging to the cyclotomic field of dimension $N$ to the one of finding a generator of ideal $\mathcal{I}^+$ that belongs to the totally real subfield, whose dimension is $\frac{N}{2}$. For a more detailed presentation of this technique, see [18, Theorem 8].

Note that even though the generator is known up to unit — i.e. $(\mathbf{g} + \bar{\mathbf{g}}) \cdot \mathbf{v}$ for $\mathbf{v} \in \mathcal{U}_{\mathbb{Q}(\zeta)}$ — the generator of $\mathcal{I}$ recovered is $\mathbf{g} \cdot \mathbf{v}$. This suffices, thanks to the last reduction part, to recover a short generator.

One could wonder if working in a real field has some relevant matter with the upcoming parts of the attack. The answer is up to our knowledge negative and we are only interested in the halving of dimension. For the asymptotic complexity, this initial reduction is somehow not meaningful since it only gives a speedup of a constant factor in the exponent. But in practice, it allows to double the dimension of the tractable cases, implying tackling security parameters twice bigger!

### 4.2 Step 2: Descent phase

Let momentarily set aside the algebraic integer obtained in the previous phase and only focus on the ideal $\mathcal{I}^+$. By construction, it is principal and generated by $\mathbf{g} + \bar{\mathbf{g}}$. From now on, all the computations are performed in the totally real subfield of dimension $\frac{N}{2}$, and from then on $N$ becomes $\frac{N}{2}$.

The goal of this phase is to find an integer $\mathbf{h}$ and a $B$-smooth principal ideal $\mathcal{I}^s$, such that $\langle \mathbf{h} \rangle = \mathcal{I}^+ \cdot \mathcal{I}^s$, for a certain bound $B > 0$. These objects are discovered recursively, by generating at each step ideals of norm smaller and smaller. Because we want a global complexity in $L_{|\Delta_{\mathbb{K}}|}(1/2)$, the smoothness bound $B$ is chosen[11] equal to $L_{|\Delta_{\mathbb{K}}|}(1/2)$. In order to bootstrap this descent, we first need to find an ideal that splits in multiple prime ideals of controlled norm, that is in our case, upper bounded by $L_{|\Delta_{\mathbb{K}}|}(1)$.

**Initial round: classical DBKZ reduction.** As announced, we aim to construct efficiently a $L_{|\Delta_{\mathbb{K}}|}(1)$-smooth principal ideal from $\mathcal{I}^+$. Formally we want to prove the following:

[11] Justification of this choice appears explicitly when we study the complexity of the descent phase of the algorithm.

**Theorem 3.** *Let $\mathbb{K}$ be a number field. Assuming Heuristicheur, from any ideal $\mathfrak{a} \subset \mathcal{O}_{\mathbb{K}}$, it is possible to generate in expected time $L_{|\Delta_{\mathbb{K}}|}(1/2)$ an integral ideal $\mathfrak{b}$ that is $L_{|\Delta_{\mathbb{K}}|}(1)$-smooth and an integer $\mathbf{v}$ such that:*

$$\langle \mathbf{v} \rangle = \mathfrak{a} \cdot \mathfrak{b}.$$

The difficulty of this preliminary part is that *a priori* the norm of the input ideal $\mathfrak{a}$ can be big. We thus want to construct at first an ideal whose norm is bounded independently from $\mathcal{N}(\mathfrak{a})$. This step relies essentially on lattice reduction. Through the canonical embedding, any integral ideal $\mathfrak{a}$ can be viewed as a Euclidean lattice. As usual when dealing with lattice reduction, we are interested in small vectors, or equivalently here, integers with small Euclidean norm. Let first study the guarantees a classical DBKZ-reduction offers on the embedding of $\mathfrak{a}$:

**Lemma 5.** *Let $\mathbb{K}$ be a number field of degree $N$, $\beta \in \{1, \ldots, N\}$ and $\mathfrak{a}$ be an ideal of $\mathcal{O}_{\mathbb{K}}$. Then it is possible to find in time $Poly(N)(3/2+o(1))^{\beta/2}$ an element $\mathbf{v} \in \mathfrak{a}$ satisfying:*

$$\|\mathbf{v}\| \leq \beta^{\frac{N}{2\beta}} \cdot |\Delta_{\mathbb{K}}|^{\frac{1}{2N}} \cdot \mathcal{N}(\mathfrak{a})^{\frac{1}{N}}$$

*where $\|.\|$ denotes the Euclidean norm.*

*Proof.* This is only a direct application of Theorem 1 and Lemma 3. Indeed, let $\mathbf{v}$ be the short vector output by DBKZ applied on the lattice of the embedding of $\mathfrak{a}$. It has determinant $\mathcal{N}(\mathfrak{a})\sqrt{|\Delta_{\mathbb{K}}|}$, yielding the announced upper bound.

Since the ideal $\mathfrak{a}$ contains $\langle \mathbf{v} \rangle$, there exists a unique integral ideal $\mathfrak{b}$ satisfying $\langle \mathbf{v} \rangle = \mathfrak{a} \cdot \mathfrak{b}$. From the guarantees on $\|\mathbf{v}\|$, we can bound the norm of this new ideal $\mathfrak{b}$:

**Corollary 3.** *With the same notations of Lemma 5, we have*

$$\mathcal{N}(\mathfrak{b}) \leq \beta^{\frac{N^2}{2\beta}} \cdot \sqrt{|\Delta_{\mathbb{K}}|}$$

*Proof.* From Lemma 5 we have:

$$\|\mathbf{v}\| \leq \beta^{\frac{N}{2\beta}} \cdot |\Delta_{\mathbb{K}}|^{\frac{1}{2N}} \cdot \mathcal{N}(\mathfrak{a})^{\frac{1}{N}}.$$

Thus, its field norm is below the $N$-th power of this bound – the $N^N$ term is negligible here – and so:

$$\mathcal{N}(\langle \mathbf{v} \rangle) \leq \beta^{\frac{N^2}{2\beta}} \cdot \sqrt{|\Delta_{\mathbb{K}}|} \cdot \mathcal{N}(\mathfrak{a}).$$

As a consequence, since $\langle \mathbf{v} \rangle = \mathfrak{a} \cdot \mathfrak{b}$, we have by the multiplicative property of the norm $\mathcal{N}(\mathfrak{b}) \leq \beta^{\frac{N^2}{2\beta}} \cdot \sqrt{|\Delta_{\mathbb{K}}|}$.

*Remark 1.* Because $\mathbb{K}$ is a cyclotomic field, we may choose a block-size $\beta$ in $\log L_{|\Delta_{\mathbb{K}}|}(1/2)$ since $\log L_{|\Delta_{\mathbb{K}}|}(1/2) = N^{1/2+o(1)} \leq N$. Then Corollary 3 generates in time $L_{|\Delta_{\mathbb{K}}|}(1/2)$ an integral ideal of norm bounded by $L_{|\Delta_{\mathbb{K}}|}(3/2)$ .

This last result allows us to find an ideal of norm bounded independently from $\mathcal{N}(\mathfrak{a})$. We then want this fresh ideal to split in multiple prime ideals of controlled norms. Thanks to Corollary 2, the probability of an integral ideal $\mathfrak{b}$ of norm bounded by $L_{|\Delta_{\mathbb{K}}|}(3/2)$ to be $L_{|\Delta_{\mathbb{K}}|}(1)$-smooth is greater than $L_{|\Delta_{\mathbb{K}}|}(1/2)^{-1}$. In addition, using ECM for testing smoothness keeps the complexity in $L_{|\Delta_{\mathbb{K}}|}(1/2)$. The analysis of this part is let for Section 4.5. Therefore, repeating the last construction $L_{|\Delta_{\mathbb{K}}|}(1/2)$ times on randomized independent inputs eventually yields a $L_{|\Delta_{\mathbb{K}}|}(1)$-smooth ideal. The simplest strategy to perform this randomization of the input ideal, is to compose it with some factors of small norms, below the bound $B = L_{|\Delta_{\mathbb{K}}|}(1/2)$. Formally we denote by $\mathcal{B} = \{\mathfrak{p}_1, \ldots, \mathfrak{p}_{|\mathcal{B}|}\}$ the set of all prime ideals of norm upper bounded by $L_{|\Delta_{\mathbb{K}}|}(1/2)$. Let $k, A > 0$ be fixed integers. We choose $\mathfrak{p}_{j_1}, \ldots, \mathfrak{p}_{j_k}$ prime ideals of norm $L_{|\Delta_{\mathbb{K}}|}(1/2)$. Then for any $k$-uple $(e_1, \ldots, e_k) \in \{1, \ldots, A\}^k$, we have:

$$\mathcal{N}(\mathfrak{a} \cdot \prod_{i=1}^{k} \mathfrak{p}_{j_i}^{e_i}) \leq \mathcal{N}(\mathfrak{a}) \cdot \prod_{i=1}^{k} \mathcal{N}(\mathfrak{p}_{j_i})^{e_i} \leq \mathcal{N}(\mathfrak{a}) \cdot L_{|\Delta_{\mathbb{K}}|}(1/2)^{k \cdot A} = \mathcal{N}(\mathfrak{a}) \cdot L_{|\Delta_{\mathbb{K}}|}(1/2)\,.$$

We know from the Landau prime ideal theorem [24] that in every number field $\mathbb{K}$, the number of prime ideals of norm bounded by $X$, denoted $\pi_{\mathbb{K}}(X)$ satisfies

$$\pi_{\mathbb{K}}(X) \sim \frac{X}{\log X}. \tag{1}$$

Thus, the randomization can be done by choosing uniformly at random the tuple $(e_1, \ldots, e_k)$ and $k$ prime ideals in $\mathcal{B}$. Since $|\mathcal{B}| = L_{|\Delta_{\mathbb{K}}|}(1/2)$, set of possible samples is large enough for our purposes.

Other ways to perform the randomization may be by randomizing directly the lattice reduction algorithm or by enumerating points of the lattice of norm close to the norm guarantee and change the basis vectors by freshly enumerated ones. This latter would be useful in practice as it reduces the number of reductions.

This last remark concludes the proof of Theorem 3. The full outline of this bootstrap section is given in Figure 3.

---

1. CurrentIdeal $\leftarrow \mathfrak{a}$.
2. **While** CurrentIdeal is not $L_{|\Delta_{\mathbb{K}}|}(1) - smooth$ **do**:
3. Choose $\mathfrak{p}_1, \cdots, \mathfrak{p}_k$ uniformly at random in $\mathcal{B}$.
4. $\mathfrak{c} \leftarrow \mathfrak{a} \cdot \prod_{1 \leq i \leq F} \mathfrak{p}_{j_i}^{e_i}$ for random $e_i \in \{1, \ldots, A\}$.
5. Generate $\mathfrak{b}$ from $\mathfrak{c}$ as in Lemma 3.
6. CurrentIdeal $\leftarrow \mathfrak{b}$.
7. **End while**
8. **Return** CurrentIdeal.

---

Fig. 3: First reduction to a $L_{|\Delta_{\mathbb{K}}|}(1)$-smooth ideal.

**Interlude: reduction with Cheon's trick.** In the proof of Theorem 3, we use the *classical*-DBKZ reduction in order to find a short element in the embedding

of the considered ideal. We could not use directly the Cheon's trick here since the norm of the ideal $\mathcal{I}^+$ – and so the determinant of its coefficient embedding – is potentially large. Nonetheless, the norm of prime ideals appearing in the factorization are by construction bounded, hence a natural question is to look at the guarantees offered when applying the sub-cited trick. The systematic treatment of this question is the aim of Theorem 4.

**Theorem 4.** *Let $\mathfrak{a}$ an integral ideal of norm bounded by $L_{|\Delta_\mathbb{K}|}(\alpha)$, for $\frac{1}{2} \leq \alpha \leq 1$. Then in expected time $L_{|\Delta_\mathbb{K}|}(1/2)$ it is possible to construct an $L_{|\Delta_\mathbb{K}|}((2\alpha+1)/4)$-smooth ideal $\mathfrak{b}$ and an algebraic integer $\mathbf{v}$ such that:*

$$\langle \mathbf{v} \rangle = \mathfrak{a} \cdot \mathfrak{b}.$$

*Proof.* The core of the proof is somehow similar to the proof of Theorem 3 as it heavily relies on lattice reduction and randomization techniques. Nonetheless, the major difference is on the embedding with respect to which the reduction is performed. In Theorem 3, the canonical embedding is used, whereas we use here the coefficient embedding $\mathcal{C}$. It avoids the apparition of a power of the discriminant in the field norm of the output of DBKZ. Nonetheless, remark that since we work in the totally real subfield, we can't use a naive coefficients embedding of this subfield. In order to benefit from the nice shape of the defining polynomial $X^N + 1$ of the cyclotomic field, we use instead a fold-in-two strategy: the embedding of $\mathcal{O}_\mathbb{K}^+$ is defined as the coefficient embedding $\mathcal{C}^+$ for the $\mathbb{Z}$-base $(\zeta^i + \zeta^{-i})_i$. Let us denote by $\|.\|_{\mathcal{C}^+}$ the induced norm. Hence, for any $v \in \mathcal{O}_\mathbb{K}^+$:

$$\|v\|_\mathcal{C} = \sqrt{2}\|v\|_{\mathcal{C}^+}.$$

Let $\mathcal{L} = \mathcal{C}^+(\mathfrak{a})$ the embedding of $\mathfrak{a}$. Its covolume is by definition its index in $\mathbb{Z}^n$, that is the index of $\mathfrak{a}$ as $\mathbb{Z}$-modules in $\mathcal{O}_\mathbb{K}^+$, which is $\mathcal{N}(\mathfrak{a})$. Then with the same block-size $\beta = \log L_{|\Delta_\mathbb{K}|}(1/2) = \mathcal{O}(\sqrt{N}\log(N))$, we have

$$\mathrm{Vol}(\mathcal{L}) \leq L_{|\Delta_\mathbb{K}|}(\alpha) = 2^{N^{\alpha+o(1)}} \leq \beta^{\frac{N^2}{2\beta}}.$$

Using the Approx-SVP algorithm of Theorem 2 yields in time $L_{|\Delta_\mathbb{K}|}(1/2)$ an integer $\mathbf{v}$ satisfying:

$$\|\mathbf{v}\|_{\mathcal{C}^+} \leq \beta^{(1+o(1))\sqrt{2\frac{\log_\beta(\det(\mathcal{L}))}{\beta}}} \leq \beta^{(1+o(1))\sqrt{4\frac{N^\alpha}{\sqrt{N}\log N}}} = L_{|\Delta_\mathbb{K}|}(\alpha/2 - 1/4).$$

Using Corollary 1 to fall back on the field norm induces:

$$\mathcal{N}_{\mathbb{K}/\mathbb{Q}}(\mathbf{v}) \leq \left(\sqrt{2}(N+1)\right)^N \cdot \|\mathbf{v}\|_\mathcal{C}^N = L_{|\Delta_\mathbb{K}|}(1) \cdot L_{|\Delta_\mathbb{K}|}(\alpha/2 + 3/4).$$

Since $\alpha \geq 1/2$, we then have $\mathcal{N}(\langle \mathbf{v} \rangle) = \mathcal{N}_{\mathbb{K}/\mathbb{Q}}(\mathbf{v}) \leq L_{|\Delta_\mathbb{K}|}(\alpha/2 + 3/4)$.

Because the ideal $\mathfrak{a}$ contains $\langle \mathbf{v} \rangle$, there exists a unique ideal $\mathfrak{b}$, satisfying $\langle \mathbf{v} \rangle = \mathfrak{a} \cdot \mathfrak{b}$. We get that $\mathcal{N}(\mathfrak{b}) \leq L_{|\Delta_\mathbb{K}|}(\alpha/2 + 3/4)$ from the multiplicative property of the norm and $\mathcal{N}(\mathfrak{a}) = L_{|\Delta_\mathbb{K}|}(1) \leq L_{|\Delta_\mathbb{K}|}(\alpha/2 + 3/4)$. Under Heuristic 1, this ideal is $L_{|\Delta_\mathbb{K}|}(\alpha/2 + 1/4)$-smooth with probability $L_{|\Delta_\mathbb{K}|}(1/2)$. Eventually performing the randomization-and-repeat technique as in the initial round, this reduction in the coefficient embedding yields the desired couple $(\mathbf{v}, \mathfrak{b})$ in expected time $L_{|\Delta_\mathbb{K}|}(1/2)$.

**Descending to $B$-smoothness.** After the first round, we end up with an $L_{|\Delta_{\mathbb{K}}|}(1)$-smooth ideal, denoted by $\mathcal{I}^{(0)}$, and an algebraic integer $\mathbf{h}^{(0)}$ satisfying

$$\langle \mathbf{h}^{(0)} \rangle = \mathcal{I}^{+} \cdot \mathcal{I}^{(0)},$$

with $\mathcal{I}^{+}$ the ideal of the totally real subfield obtained after phase 4.1. The factorization of $\mathcal{I}^{(0)}$ gives

$$\mathcal{I}^{(0)} = \prod_j \mathcal{I}_j^{(0)},$$

where the $\mathcal{I}_j^{(0)}$ are integral ideals of norm upper bounded by $L_{|\Delta_{\mathbb{K}}|}(1)$. Taking the norms of the ideals involved in this equality ensures that the number of terms in this product is $\mathcal{O}(n_{\mathcal{I}})$, with $n_{\mathcal{I}} = \frac{\log|\Delta_{\mathbb{K}}|}{\log\log|\Delta_{\mathbb{K}}|} = \mathcal{O}(N)$. Then applying Theorem 4 on each small ideal $\mathcal{I}_j^{(0)}$ gives rise to ideals $\mathcal{I}_j^{(1)}$ in expected time $L_{|\Delta_{\mathbb{K}}|}(1/2)$ that are $L_{|\Delta_{\mathbb{K}}|}\left(\frac{2\times 1+1}{4}\right) = L_{|\Delta_{\mathbb{K}}|}(3/4)$-smooth and integers $\mathbf{h}_j^{(1)}$ such that for every $j$,

$$\langle \mathbf{h}_j^{(1)} \rangle = \mathcal{I}_j^{(0)} \cdot \mathcal{I}_j^{(1)}.$$

For each factor $\mathcal{I}_j^{(1)}$, let us write its decomposition:

$$\mathcal{I}_j^{(1)} = \prod_k \mathcal{I}_{j,k}^{(1)}.$$

Once again, the number of terms appearing is $\mathcal{O}(n_{\mathcal{I}})$. Because we have the inequality $\mathcal{N}(\mathcal{I}_{j,k}^{(1)}) \leq L_{|\Delta_{\mathbb{K}}|}(3/4)$, then performing the same procedure on each ideal $\mathcal{I}_{j,k}^{(1)}$ now yields $L_{|\Delta_{\mathbb{K}}|}(5/8)$-smooth ideals $\mathcal{I}_{j,k}^{(2)}$ and integers $\mathbf{h}_{j,k}^{(2)}$ such that

$$\langle \mathbf{h}_{j,k}^{(2)} \rangle = \mathcal{I}_{j,k}^{(1)} \cdot \mathcal{I}_{j,k}^{(2)},$$

once again in expected time $L_{|\Delta_{\mathbb{K}}|}(1/2)$. Remark that this smoothness bound in $L_{|\Delta_{\mathbb{K}}|}(5/8)$ is obtained as $L_{|\Delta_{\mathbb{K}}|}\left(\frac{2\times 3/4+1}{4}\right)$, as exposed in Theorem 4. This reasoning naturally leads to a recursive strategy for reduction. At step $k$, we want to reduce an ideal $\mathcal{I}_{a_1,\ldots,a_{k-1}}^{(k-1)}$ which is $L_{|\Delta_{\mathbb{K}}|}\left(1/2 + 1/2^{k+1}\right)$-smooth. As before, we have a decomposition – in $\mathcal{O}(n_{\mathcal{I}})$ terms – in smaller ideals:

$$\mathcal{I}_{a_1,\ldots,a_{k-1}}^{(k-1)} = \prod_j \mathcal{I}_{a_1,\ldots,a_{k-1},j}^{(k-1)}.$$

Using Theorem 4 on each factor $\mathcal{I}_{a_1,\ldots,a_{k-1},j}^{(k-1)}$ which have norm bounded by $L_{|\Delta_{\mathbb{K}}|}\left(1/2 + 1/2^{k+1}\right)$, leads to $L_{|\Delta_{\mathbb{K}}|}\left(1/2 + 1/2^{k+2}\right)$-smooth ideals $\mathcal{I}_{a_1,\ldots,a_{k-1},j}^{(k)}$ and algebraic integers $\mathbf{h}_{a_1,\ldots,a_{k-1},j}^{(k)}$ such that

$$\langle \mathbf{h}_{a_1,\ldots,a_{k-1},j}^{(k)} \rangle = \mathcal{I}_{a_1,\ldots,a_{k-1},j}^{(k-1)} \cdot \mathcal{I}_{a_1,\ldots,a_{k-1},j}^{(k)},$$

since $\frac{2\times(1/2+1/2^{k+1})+1}{4} = 1/2 + 1/2^{k+2}$.

As a consequence, one can generate $L_{|\Delta_{\mathbb{K}}|}(1/2 + 1/\log N)$-smooth ideals in the previous reasoning in at most $\lceil \log_2(\log N) \rceil$ recursive steps. At this point only $(n_{\mathcal{I}})^{\lceil \log_2(\log N) \rceil}$ ideals and algebraic integers are appearing since at each step this number is multiplied by a factor $\mathcal{O}(n_{\mathcal{I}})$. As deriving one couple integer/ideal is done in expected time $L_{|\Delta_{\mathbb{K}}|}(1/2)$, the whole complexity remains in $L_{|\Delta_{\mathbb{K}}|}(1/2)$.

However, as $|\Delta_{\mathbb{K}}| = N^N$, a quick calculation entails that

$$\begin{aligned}\log L_{|\Delta_{\mathbb{K}}|}\left(\frac{1}{2} + \frac{1}{\log(N)}\right) &= \mathcal{O}(N^{\frac{1}{2}+\frac{1}{\log N}}\log(N)) \\ &= \mathcal{O}(N^{\frac{1}{2}}\log(N)) \cdot N^{\frac{1}{\log N}}.\end{aligned}$$

Since the last factor is $e = \exp(1)$, we obtain that

$$\log L_{|\Delta_{\mathbb{K}}|}\left(\frac{1}{2} + \frac{1}{\log(N)}\right) = L_{|\Delta_{\mathbb{K}}|}\left(\frac{1}{2}\right),$$

so that after at most $\lceil \log_2(\log N) \rceil$ steps, we have ideals that are $L_{|\Delta_{\mathbb{K}}|}(1/2)$-smooth.

At the end of this final round, we may express the input ideal as the product of ideals for which we know a generator and others that have by construction norms bounded by $L_{|\Delta_{\mathbb{K}}|}(1/2)$. Let denote $\mathcal{K}$ the final step. For avoiding to carry inverse ideal, we may assume without loss of generality[12] that $\mathcal{K}$ is even. Explicitly we have:

$$\begin{aligned}\langle \mathbf{h}^{(\mathbf{0})} \rangle &= \mathcal{I}^+ \cdot \mathcal{I}^{(0)} \\ &= \mathcal{I}^+ \cdot \prod_{a_1} \mathcal{I}_{a_1}^{(0)} \\ &= \mathcal{I}^+ \cdot \left\langle \frac{\prod_{a_1} \mathbf{h}_{a_1}^{(1)} \prod_{a_1,a_2,a_3} \mathbf{h}_{a_1,a_2,a_3}^{(3)}}{\prod_{a_1,a_2} \mathbf{h}_{a_1,a_2}^{(2)}} \right\rangle \cdot \prod_{a_1,a_2,a_3} \mathcal{I}_{a_1,a_2,a_3}^{(3)} \\ &= \mathcal{I}^+ \cdot \left\langle \prod_{a_1,\ldots,a_{\mathcal{K}+1}} \frac{\prod_{t\in 2\mathbb{Z}+1} \mathbf{h}_{a_1,\ldots,a_t}^{(t)}}{\prod_{s\in 2\mathbb{Z}} \mathbf{h}_{a_1,\ldots,a_s}^{(s)}} \right\rangle \cdot \underbrace{\prod_{a_1,\ldots,a_{\mathcal{K}+1}} \mathcal{I}_{a_1,\ldots,a_{\mathcal{K}+1}}^{(\mathcal{K})}}_{:=\mathcal{I}^s}.\end{aligned}$$

In this last expression, the indices are chosen such that $1 \leq t \leq \mathcal{K}$ and $2 \leq s \leq \mathcal{K}$. We also recall that all the quantities involved here belong to the totally real subfield $\mathbb{Q}(\zeta + \zeta^{-1})$.

[12] We can always run an additional step in the descent without changing the whole complexity.

By construction, $\mathcal{I}^s$ is $L_{|\Delta_{\mathbb{K}}|}(1/2)$-smooth and we directly get $\mathbf{h} \in \mathcal{O}_{\mathbb{K}}^+$ such that

$$\langle \mathbf{h} \rangle = \mathcal{I}^+ \cdot \mathcal{I}^s.$$

The full outline of this descent phase is sketched in Figure 4.

Remark that the number of terms, which is at most $\mathcal{O}(N)^{\mathcal{K}}$ is in $L_{|\Delta_{\mathbb{K}}|}(o(1))$, so that it is negligible in the final complexity estimate.

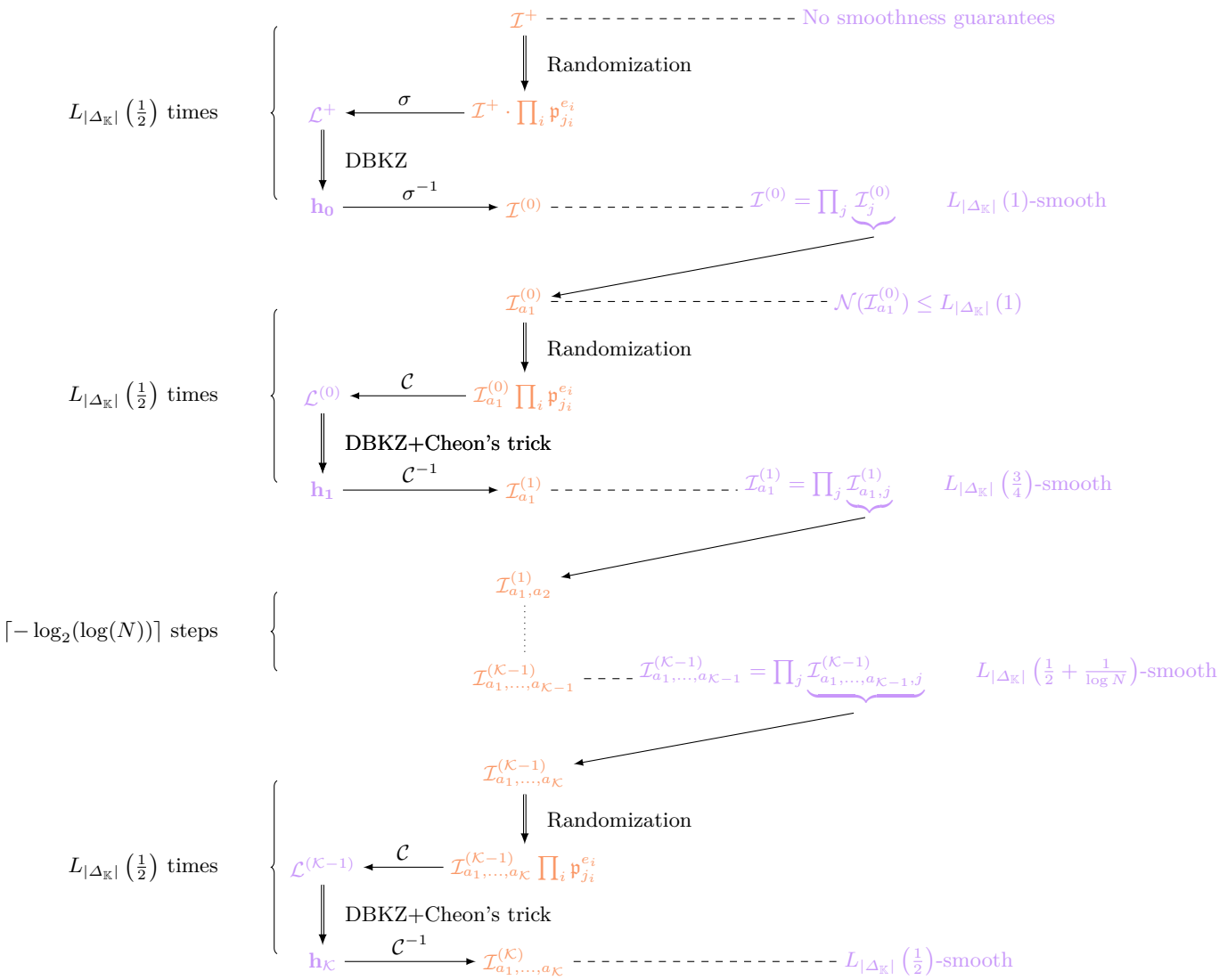


Fig. 4: The descent algorithm.

### 4.3 Step 3: Case of $L_{|\Delta_{\mathbb{K}}|}(1/2)$-smooth ideals

At this point, we have reduced our problem of finding a generator for a large ideal to a smaller one: find a generator for a principal ideal which is $L_{|\Delta_{\mathbb{K}}|}(1/2)$-smooth. If we can find a generator of this ideal $\mathcal{I}^s$ in time $L_{|\Delta_{\mathbb{K}}|}(1/2)$, from the previous steps we directly recover the generator of $\mathcal{I}^+$, and so the generator of $\mathcal{I}$, that is the secret key. The idea to tackle the final problem is inspired from class group computation: we consider the previously introduced set $\mathcal{B}$ of prime ideals of norm below $L_{|\Delta_{\mathbb{K}}|}(1/2)$ and look for relations of the shape

$$(\mathbf{v}) = \prod_i \mathfrak{p}_i^{e_i}, \qquad \text{for} \quad \mathbf{v} \in \mathcal{O}_{\mathbb{K}}.$$

Actually, because the problem has firstly been reduced to the totally real subfield (Step 1), it suffices to only consider relations in $\mathcal{O}_{\mathbb{K}^+}$.

As the classes of prime ideals in $\mathcal{B}$ generates the class group $\mathrm{Cl}(\mathcal{O}_{\mathbb{K}})$ (see [1]), we have a surjective morphism:

$$\begin{array}{ccccc} \mathbb{Z}^{|\mathcal{B}|} & \xrightarrow{\phi} & \mathcal{S}_{\mathcal{I}} & \xrightarrow{\pi} & \mathrm{Cl}(\mathcal{O}_{\mathbb{K}}) \\ (e_1, \cdots, e_{|\mathcal{B}|}) & \longmapsto & \prod_i \mathfrak{p}_i^{e_i} & \longmapsto & \prod_i [\mathfrak{p}_i]^{e_i}, \end{array}$$

Formally, a relation is then an element of $\mathrm{Ker}\,(\phi \circ \pi)$. Thanks to class group studies (see for instance [10,6]), the relations form a full-rank sublattice of $\mathbb{Z}^{|\mathcal{B}|}$. Hence we need to find at least $L_{|\Delta_{\mathbb{K}}|}(1/2)$ linearly independent relations to generate this lattice. The relation collection is performed in a similar way as [4]: due to the good shape of the defining polynomial $X^N + 1$, the algebraic integers whose representation as polynomials in $\zeta$ have small coefficients, also have small norms.

Let us fix an integer $0 < A \leq L_{|\Delta_{\mathbb{K}}|}(0) = \log |\Delta_{\mathbb{K}}|$. Then for any integers $(v_0, \cdots, v_{\frac{N}{2}-1}) \in \{-A, \ldots, A\}^{\frac{N}{2}}$, we define the element $\mathbf{v} = v_0 + \sum_{i \geq 1} v_i \left(\zeta^i + \zeta^{-i}\right)$. The norm of this element in the totally real subfield is upper bounded by $L_{|\Delta_{\mathbb{K}}|}(1)$. Indeed, it corresponds to the square root of its norm in $\mathcal{O}_{\mathbb{K}}$, which is below $N^N \cdot A^N = L_{|\Delta_{\mathbb{K}}|}(1)$ by Lemma 2. Then under Heuristic 1, the element $\mathbf{v}$ generates an ideal $\langle \mathbf{v} \rangle$ that is $L_{|\Delta_{\mathbb{K}}|}(1/2)$-smooth with probability $L_{|\Delta_{\mathbb{K}}|}(1/2)^{-1}$. As such considering independently drawn $L_{|\Delta_{\mathbb{K}}|}(1/2)$ algebraic integers is sufficient to yield one relation.

Assuming that the relations obtained by this method are independent, we then rely on a commonly used heuristic in the area of class group computation to bound the required number of relations.

**Heuristic 3** *There exists $Q$ negligible with respect to $|\mathcal{B}|$ such that collecting $Q \cdot |\mathcal{B}|$ relations suffices to generate the whole lattice of relations.*

Thanks to Equation (1), we know that $\mathcal{B}$ contains about $L_{|\Delta_{\mathbb{K}}|}(1/2)$ elements. Therefore, $L_{|\Delta_{\mathbb{K}}|}(1/2)$ relations are needed thanks to Heuristic 3, implying that $L_{|\Delta_{\mathbb{K}}|}(1/2)^2 = L_{|\Delta_{\mathbb{K}}|}(1/2)$ independently drawn algebraic integers suffice to

generate the whole lattice of relations. Of course, the set of integers arising from the previous construction is large enough to allow such repeated sampling, because its size is $L_{|\Delta_{\mathbb{K}}|}(1)$. We store the the relations in a $|\mathcal{B}| \times Q|\mathcal{B}|$ matrix $M$, as well as the corresponding algebraic integers in a vector $G$.

$$\left.\begin{array}{rc} M & \begin{pmatrix} e_{1,1} & \cdots & e_{1,i} & \cdots & e_{1,Q|\mathcal{B}|} \\ e_{2,1} & \cdots & e_{2,i} & \cdots & e_{2,Q|\mathcal{B}|} \\ \vdots & & \vdots & & \vdots \\ e_{|\mathcal{B}|,1} & \cdots & e_{|\mathcal{B}|,i} & \cdots & e_{|\mathcal{B}|,Q|\mathcal{B}|} \end{pmatrix} \\ \hline G & \begin{pmatrix} \mathbf{v}_1 & \cdots & \mathbf{v}_i & \cdots & \mathbf{v}_{|\mathcal{B}|} \end{pmatrix} \end{array}\right\} \quad \forall i, (\mathbf{v}_i) = \prod_{j=0}^{|\mathcal{B}|} \mathfrak{p}_i^{e_{j,i}}.$$

The $L_{|\Delta_{\mathbb{K}}|}(1/2)$-smooth ideal $\mathcal{I}^s$ splits over the set $\mathcal{B}$, so that there exists a vector $Y$ of $\mathbb{Z}^{|\mathcal{B}|}$ containing the exponents of the factorization

$$\mathcal{I}^s = \prod_i \mathfrak{p}_i^{Y_i}$$

As the relations stored in $M$ generate the lattice of all elements of this form, the vector $Y$ necessarily belongs to it. Hence solving the equation $MX = Y$ yields a vector $X \in \mathbb{Z}^{Q|\mathcal{B}|}$ from which we can recover a generator of the ideal since:

$$\prod_i \mathfrak{p}_i^{Y_i} = \langle \mathbf{v}_1^{X_1} \cdots \mathbf{v}_{Q|\mathcal{B}|}^{X_{Q|\mathcal{B}|}} \rangle.$$

By construction, $\mathcal{N}(\mathcal{I}^s) \leq L_{|\Delta_{\mathbb{K}}|}(\mathcal{K}/2 + 1/2)$ so that the coefficients of $Y$ are below $L_{|\Delta_{\mathbb{K}}|}(0)$. Since solving such a linear system can be done in time $\mathrm{Poly}(d, \log \|M\|)$ where $d$ is the dimension of the matrix and $\|M\| = \max |M_{i,j}|$ the maximum of its coefficients, we are able to recover $X$ with a complexity in $L_{|\Delta_{\mathbb{K}}|}(1/2)$.

### 4.4 Final Step: Reduction to a short generator

As mentioned in Section 3, this part of the algorithm is a result of Cramer, Ducas, Peikert and Regev [16]. They state that recovering a short generator from an arbitrary one can be solved in polynomial time, in any cyclotomic ring of prime-power index. For completeness purposes, we give here a brief overview of this reduction.

As a liminary observation, note that for those fields, a set of fundamental units is given for free, whereas their computation in arbitrary number fields is computationally hard. A second remark is that we get the promise that there exists a small generator of the considered ideal. Then, instead of solving a general *closest vector problem* (CVP), we solve an instance of *bounded-distance decoding* (BDD) problem. The key argument is based on a precise study of the geometry of the log-unit lattice of prime-power cyclotomic fields (see Appendix A.3 for

basic recalls about this lattice). Finally, their geometric properties make possible to solve BDD in this lattice in polynomial time, instead of exponential time as for generic instances.

**Theorem 5 ([16, Theorem 4.1]).** *Let $D$ be a distribution over $\mathbb{Q}(\zeta)$ with the property that for any tuple of vectors $\mathbf{v}_1, \cdots, \mathbf{v}_{N/2-1} \in \mathbb{R}^{N/2-1}$ of Euclidean norm 1 that are orthogonal to the all-1 vector $\mathbf{1}$, the probability that $|(\mathrm{Log}(\mathbf{g}), \mathbf{v_i})| < c\sqrt{2N} \cdot \log(2N)^{-3/2}$ holds for all $i$ is at least some $\alpha > 0$, where $\mathbf{g}$ is chosen from $D$ and $c$ is a universal constant. Then there is an efficient algorithm that given $\mathbf{g}' = \mathbf{g} \cdot \mathbf{u}$, where $\mathbf{g}$ chosen from $D$ and $\mathbf{u} \in C$ is a cyclotomic unit, outputs an element of the form $\zeta^j \cdot \mathbf{g}$ with probability at least $\alpha$.*

The reader might argue that in order to use this theorem on the output of our algorithm we should ensure that we recover a generator up to a *cyclotomic unit* and not up to an arbitrary unit. In the specific case of power-of-two cyclotomic fields, we can rely on Weber's heuristic 2 to ensure this constraint. In case $h^+(N) > 1$, two solutions are given by Cramer, Ducas, Peikert and Regev [16]. The first one is to directly compute the group of units, which is hopefully determined by the kernel of the matrix $M$ arising in the third stage[13]. One can then enumerate the $h^+(N)$ classes of the group of units modulo the subgroup of cyclotomic units. Another possibility is to generate a list of ideals, sampled according to the same distribution as the input ideal, with a known generator. Then, we run the PIP algorithm on these ideals, and deduce the cosets of the group of units modulo the subgroup of cyclotomic units, which are likely to be output.

The whole key recovery, combining our PIP algorithm and the aforementioned reduction is outlined in Figure 5.

1. Compute a generator $\mathbf{g_0}$ of $\mathcal{I}$ with Gentry-Szydlo, descent and relation collection.
2. Let $\mathbf{B}$ the basis defined by the $\mathrm{Log}(\mathbf{b_i})$ for $\mathbf{b}_i = \frac{\zeta_m^i - 1}{\zeta_m - 1}$.
3. Let $t = \mathrm{Log}(\mathbf{g_0}) + \mathrm{Log}(\mathcal{O}_\mathbb{K})$
4. Return Babai's rounding of $\mathbf{B}\lfloor(\mathbf{B}^\vee)^t \cdot t\rceil$.

where $\lfloor\cdot\rceil$ denotes the rounding function: $\lfloor c\rceil = \lfloor c + \frac{1}{2}\rfloor$.

Fig. 5: Recovery of the secret key by PIP+[16]

### 4.5 Complexity analysis

The whole runtime of our attack is $L_{|\Delta_\mathbb{K}|}(1/2)$, that is in $2^{N^{1/2+o(1)}}$ operations. We have already mentioned the complexity of most parts of our algorithm. However, we provide a brief summary in this paragraph to ensure the entirety of our result.

[13] Another possibility is to use the saturation method which might run in polynomial time [5].

For the reduction algorithms, DBKZ and Cheon's trick, the block-size is always in $\log L_{|\Delta_{\mathbb{K}}|}(1/2)$ so that the complexity is $L_{|\Delta_{\mathbb{K}}|}(1/2)$. Our choice for the smoothness bound $B = L_{|\Delta_{\mathbb{K}}|}(1/2)$ ensures that the step of relation collection together with the linear system solution are derived in time $L_{|\Delta_{\mathbb{K}}|}(1/2)$.

In addition, from the work of [18], we get that the first part of the algorithm, corresponding to the reduction to the totally real subfield is performed in polynomial time. Similarly the last part, which corresponds with the generation of a small generator from an arbitrary one, also runs in polynomial time, thanks to the results of [16].

We highlight now two points whose complexity were eluded in the exposition of the algorithm:

- *Arithmetic of ideals.* All the operations made on ideals are classical, with complexities polynomial in the dimension and in the size of the entries (see for instance [14, Chapter 4]), which is way below the bound of $L_{|\Delta_{\mathbb{K}}|}(1/2)$.
- *Smoothness tests.* The strategy is to deal with the norms of ideals, that are integers. The largest norm arising in the computations is in $L_{|\Delta_{\mathbb{K}}|}(3/2)$ and appears after the initial DBKZ reduction. Testing $L_{|\Delta_{\mathbb{K}}|}(1)$-smoothness for an integer of this size is easier than completely factorizing it, even if both methods share the same asymptotic complexity in $L_{|\Delta_{\mathbb{K}}|}(1/2)$[14]. Hence all the smoothness tests performed have complexity dominated by $L_{|\Delta_{\mathbb{K}}|}(1/2)$.

As a consequence the global complexity is given by the first and last steps of the descent, that is in $L_{|\Delta_{\mathbb{K}}|}(1/2)$.

*Remark 2.* This algorithm has a complexity in $L_{|\Delta_{\mathbb{K}}|}(1/2)$ in the discriminant, that represents the size of the number field involved. However, it is important to figure out that the parameters of the keys have $N^{3/2}$ bits. Therefore we present an algorithm that is "sort of" $L(1/3)$ in the size of the inputs.

## 5 Implementation results

In addition to the theoretical improvement, our algorithm permits in practice to break concrete cryptosystems. Our discussion is based on the scheme presented by Smart and Vercauteren at PKC 2010. In [39, Section 7], security estimations are given for parameters $N = 2^n$ for $8 \leq n \leq 11$ since they are unable to generate keys for larger parameters. Our implementation allows us to recover the secret key from the public key for $N = 2^8 = 256$ in less than a day. The code runs with `PARI-GP` [35], with an external call to `fplll` [17], and all the computations are performed on a Intel(R) Xeon(R) CPU E3-1275 v3 @ 3.50GHz with 32Go of memory. Indeed the Gentry-Szydlo algorithm requires large storage.

We perform the key generation as recalled in Figure 2. We then obtain a generator for the ideal as a polynomial in $\zeta = \zeta_{512}$, of degree 255 and coefficients absolutely bounded by $2^{\sqrt{256}} + 1 = 65537$. That corresponds to ideals whose norm

[14] Factorizing an integer $N$ is done in $L_N(1/3)$.

has about 4800 bits in average, that is below the bound 6145 from Lemma 2, but above the size given in [39] (4096). As for every timing arising in this section, we have derived a set of 10 keys, and the given time is the average one. Thus, deriving a secret key takes on average 30 seconds. We test 1381 algebraic integers for finding 10 having prime norm. Then the public key is derived from the secret key in about 96 seconds.

While in theory, the first reduction to the totally real subfield seems to be of limited interest, it is clearly the main part of the practical results: indeed it reduces in our example the size of the matrices involved from $256 \times 256$ to $128 \times 128$. As we know that lattice-reduction is getting worse while the dimension grows, this part is the key point of the algorithm. Our code essentially corresponds to the Gentry-Szydlo algorithm together with the trick explained in Section 4.1, in order to output the element $\mathbf{u}$ and a basis of the ideal $\mathcal{I}^+$ generated by $\mathbf{g} + \bar{\mathbf{g}}$. This part of the algorithm has the largest runtime, about 20 hours, and requires 24Go of memory.

At this point, we put aside $\mathbf{u}$ and only consider the ideal $\mathcal{I}^+$. Our goal is to recover one generator of this ideal, and a multiplication with $\frac{1}{1+\mathbf{u}}$ is going to lead to the generator of the input ideal. The method we have presented is to reduce step by step the norm of the ideals involved by performing lattice reductions. However we observe that for the cases we run, the first reduction suffices: the short vector we find corresponds with the generator. We make use of the BKZ algorithm implemented in `fplll` [17], with block-size 24 to begin. It gives a correct generator with probability higher than 0.75 and runs in less than 10 minutes. If the output is not correct, we increase the block-size to 30. This always works and requires between 2 and 4 hours.

In addition to the good behavior of this reduction, the generator we exhibit is already small, by construction. More precisely, it corresponds to $\mathbf{g} + \bar{\mathbf{g}}$, up to a factor that is a power of $\zeta$. Hence, we recover $\mathbf{g} \cdot \zeta^i$ thanks to $\mathbf{u}$ and the algorithm analyzed by Cramer, Ducas, Peikert and Regev is unnecessary for our concern. They key recovery is already completed after these two first steps. We still implement this part together with a method for recovering the actual private key (up to sign). Indeed, because all its coefficients are even except the constant one, it is easy to identify the power of $\zeta$ that appears as a factor during the computation.

**Additional work.** To handle the whole problem, we still look at the runtime of the other steps of the algorithm displayed in this paper. It essentially corresponds to the construction of the relation matrix.

We fix our factor base as all the prime ideals in the totally-real field that lie above a prime number $p$ that is below the bound $c\,(\log|\Delta_{\mathbb{K}}|)^2$, for $c \in \{0.1, 0.2, 0.3\}$. We give in Table 1 the values, together with the size of the factor base and the time required for building it in `MAGMA` [15]. The computations are performed on a laptop with Intel(R) Core(TM) i7-4710MQ CPU @ 2.50GHz and 8Go of RAM for this part.

Naturally, this choice of bound would not be sufficient for the descent described above, because it is polynomial and not subexponential. However, it provides a relation matrix for the computation of the class group. Reaching a subexponential bound seems unlikely in that way, that supports the fact that our implementation results are consequences of the small dimension obtained by the Gentry-Szydlo algorithm.

| $c$ | Bound | #primes | #Factor Base | Time (sec) |
|---|---|---|---|---|
| 0.1 | 201516 | 149 | 18945 | 1240 |
| 0.2 | 403033 | 274 | 35073 | 2320 |
| 0.3 | 604549 | 385 | 49281 | 3320 |

Table 1: Construction of differently parametrized factor bases

The relation collection is performed using algebraic integers of the shape

$$\sum_{i=1}^{5} \zeta^{a_i} + \zeta^{-a_i} = \sum_{i=1}^{5} \zeta^{a_i} - \zeta^{256-a_i},$$

for $a_i$ chosen at random in $\{1, \ldots, 255\}$. This is inspired from the work of Miller [34]. We use C++ code with NTL Library [38] for finding a set of integers with different norms that suffice for generating the full lattice of relations (see Section 4.3). The size of these sets depends on the bound we have chosen and on the relations picked, so that the timings may vary. Our results are provided in Table 2. Once we know these integers, we use Magma for building the entire matrix of relations. In particular, we make use of the automorphisms on the field for deriving 128 relations from each integer – this is the reason we use integers of different norms. Eventually, the matrices we get are full-rank.

| $c$ | #relations | Time (hours) | |
|---|---|---|---|
| | | relation collection | matrix construction |
| 0.1 | 1500 | 8.6 | 1.7 |
| 0.2 | 3400 | 13.8 | 4.9 |
| 0.3 | 6300 | 23.9 | 10.7 |

Table 2: Relation collection for the different parameters

We also run our code for the algorithm described in [16] on inputs constructed as a secret key multiplied by a random non-zero vector of the log-unit lattice (because in the full attack described previously, we only have the null vector). This runs in 150 seconds.

To conclude, for the parameter $N = 2^8$, the time of the key recovery is below 24 hours, and the main part of the computation comes from the reduction to the totally real subfield. Hence, one may wonder if this step is mandatory, and the answer is yes, because the surprisingly good practical behavior of the BKZ reduction is a conjoint consequence of the dimension of lattices involved on the one hand — the regime for such medium dimension allows better practical output bounds than the theoretical worst case — and the specificity of the geometry of the considered ideals induced by the abnormally small norm of its generator.

## References


1. Bach, E.: Explicit bounds for primality testing and related problems. Mathematics of Computation 55, 355–380 (1990)
2. Becker, A., Ducas, L., Gama, N., Laarhoven, T.: New directions in nearest neighbor searching with applications to lattice sieving. Cryptology ePrint Archive, Report 2015/1128 (2015), http://eprint.iacr.org/2015/1128
3. Belabas, K., Diaz y Diaz, F., Friedman, E.: Small generators of the ideal class group. Mathematics of Computation 77(262), 1185–1197 (2008)
4. Biasse, J.: An L(1/3) algorithm for ideal class group and regulator computation in certain number fields. Mathematics of Computation 83, 2005–2031 (2014)
5. Biasse, J., Fieker, C.: Improved techniques for computing the ideal class group and a system of fundamental units in number fields. In: Proceedings of the 10th Algorithmic Number Theory Symposium (ANTS X) Open Book Series - Mathematical Science Publishers, 2012. vol. 1, pp. 113–133 (2012)
6. Biasse, J., Fieker, C.: Subexponential class group and unit group computation in large degree number fields. LMS Journal of Computation and Mathematics 17, 385–403 (2014)
7. Biasse, J.F.: A fast algorithm for finding a short generator of a principal ideal of $\mathbb{Q}(\zeta_{2^n})$. arXiv:1503.03107v1 (2015), http://arxiv.org/abs/1503.03107v1
8. Biasse, J., Song, F.: Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields. In: Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016. pp. 893–902 (2016)
9. Bistritz, Y., Lifshitz, A.: Bounds for resultants of univariate and bivariate polynomials. Linear Algebra and its Applications 432, 1995–2005 (2010)
10. Buchmann, J.: A subexponential algorithm for the determination of class groups and regulators of algebraic number fields. Séminaire de Théorie des Nombres, Paris 1988-1989 pp. 27–41 (1990)
11. Campbell, P., Groves, M., Shepherd, D.: SOLILOQUY: A cautionary tale. ETSI 2nd Quantum-Safe Crypto Workshop (2014), http://docbox.etsi.org/workshop/2014/201410_CRYPTO/S07_Systems_and_Attacks/S07_Groves.pdf

12. Canfield, E.R., Erdős, P., Pomerance, C.: On a problem of Oppenheim concerning 'factorisatio numerorum'. Journal of Number Theory 17, 1–28 (1983)
13. Cheon, J.H., Lee, C.: Approximate algorithms on lattices with small determinant. Cryptology ePrint Archive, Report 2015/461 (2015), http://eprint.iacr.org/2015/461
14. Cohen, H.: A course in computational algebraic number theory, Graduate Texts in Mathematics, vol. 138. Springer-Verlag, New-York (1993)
15. Computational Algebra Group, University of Sydney: MAGMA, version 2.21.6 (2016), http://magma.maths.usyd.edu.au/magma/
16. Cramer, R., Ducas, L., Peikert, C., Regev, O.: Recovering short generators of principal ideals in cyclotomic rings. In: Advances in Cryptology - EUROCRYPT 2016. Proceedings, Part II. pp. 559–585 (2016)
17. The FPLLL development team: fplll, version 5.0 (2016), https://github.com/fplll/fplll
18. Garg, S., Gentry, C., Halevi, S.: Candidate multilinear maps from ideal lattices. In: Advances in Cryptology - EUROCRYPT 2013. Proceedings. pp. 1–17 (2013)
19. Gentry, C.: Fully homomorphic encryption using ideal lattices. In: Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009. pp. 169–178 (2009)
20. Gentry, C., Szydlo, M.: Cryptanalysis of the revised NTRU signature scheme. In: Advances in Cryptology - EUROCRYPT 2002. Proceedings. pp. 299–320 (2002)
21. Hafner, J.L., McCurley, K.S.: A rigorous subexponential algorithm for computation of class groups. Journal of American Mathematical Society 2, 839–850 (1989)
22. Hoffstein, J., Pipher, J., Silverman, J.H.: NTRU: A ring-based public key cryptosystem. In: Algorithmic Number Theory, Third International Symposium, ANTS-III. Proceedings. pp. 267–288 (1998)
23. Kirchner, P.: Algorithms on ideal over complex multiplication order. Cryptology ePrint Archive, Report 2016/220 (2016), http://eprint.iacr.org/2016/220
24. Landau, E.: Neuer beweis des primzahlsatzes und beweis des primidealsatzes. Mathematische Annalen 56, 645–670 (1903)
25. Langlois, A., Stehlé, D., Steinfeld, R.: GGHLite: More efficient multilinear maps from ideal lattices. In: Advances in Cryptology - EUROCRYPT 2014. Proceedings. pp. 239–256 (2014)
26. Lenstra, A.K., Lenstra Jr., H.W., Manasse, M.S., Pollard, J.M.: The number field sieve. In: Proceedings of the 22nd Annual ACM Symposium on Theory of Computing. pp. 564–572 (1990)
27. Lenstra, H.W., Silverberg, A.: Revisiting the Gentry-Szydlo algorithm. In: Advances in Cryptology - CRYPTO 2014. Proceedings, Part I. pp. 280–296 (2014)
28. Lyubashevsky, V., Micciancio, D., Peikert, C., Rosen, A.: SWIFFT: A modest proposal for FFT hashing. In: Fast Software Encryption FSE 2008. pp. 54–72 (2008)
29. Lyubashevsky, V., Peikert, C., Regev, O.: On ideal lattices and learning with errors over rings. Journal of the ACM 60(6), 1–23 (2013)
30. Lyubashevsky, V., Peikert, C., Regev, O.: A toolkit for Ring-LWE cryptography. In: Advances in Cryptology - EUROCRYPT 2013. Proceedings. pp. 35–54 (2013)
31. Micciancio, D.: Generalized compact knapsacks, cyclic lattices, and efficient one-way functions from worst-case complexity assumptions. In: 43rd Symposium on Foundations of Computer Science (FOCS 2002). Proceedings. pp. 356–365 (2002)
32. Micciancio, D., Walter, M.: Practical, predictable lattice basis reduction. In: Advances in Cryptology - EUROCRYPT 2016. Proceedings, Part I. pp. 820–849 (2016)

33. Micciancio, D., Warinschi, B.: A linear space algorithm for computing the Hermite Normal Form. In: Proceedings of the 2001 International Symposium on Symbolic and Algebraic Computation, ISSAC 2001. pp. 231–236 (2001)
34. Miller, J.C.: Class numbers of totally real fields and applications to the weber class number problem. Acta Arithmetica 164, 381–397 (2014)
35. The PARI Group, Bordeaux: PARI/GP, version 2.7.6 (2016), http://pari.math.u-bordeaux.fr/
36. Schank, J.: LOGCVP, Pari implementation of CVP in $\log \mathbb{Z}[\zeta_{2^n}]^*$ (2015), https://github.com/jschanck-si/logcvp
37. Seysen, M.: A probabilistic factorization algorithm with quadratic forms of negative discriminant. Mathematics of Computation 84, 757–780 (1987)
38. Shoup, V.: NTL: A Library for doing Number Theory, version 9.11.0 (2016), http://http://www.shoup.net/ntl/
39. Smart, N.P., Vercauteren, F.: Fully homomorphic encryption with relatively small key and ciphertext sizes. In: Public Key Cryptography - PKC 2010. Proceedings. pp. 420–443 (2010)
40. Stehlé, D., Steinfeld, R., Tanaka, K., Xagawa, K.: Efficient public key encryption based on ideal lattices. In: Advances in Cryptology - ASIACRYPT 2009. Proceedings. pp. 617–635 (2009)
41. Thiel, C.: On the complexity of some problems in algorithmic algebraic number theory. Ph.D. thesis, Universität des Saarlandes (1995), https://www.cdc.informatik.tu-darmstadt.de/reports/reports/Christoph_Thiel.diss.pdf
42. Washington, L.C.: Introduction to Cyclotomic Fields, Graduate Texts in Mathematics, vol. 83. Springer-Verlag, New York, 2nd edn. (1997)

# A Mathematical background recalls

## A.1 Hermite normal form.

**Definition 1.** *A $m \times n$ matrix $\mathbf{B}$ with integer entries has a (unique) Hermite normal form (HNF) $\mathbf{H}$ such that there exists a square unimodular matrix $\mathbf{U}$ satisfying $\mathbf{H} = \mathbf{BU}$ and*

1. *$\mathbf{H}$ is lower triangular, $h_{i,j} = 0$ for $i < j$, and any columns of zeros are located on the right.*
2. *The leading coefficient (the first nonzero entry from the top, also called the* pivot*) of a nonzero column is always strictly below of the leading coefficient of the column before it and is positive.*
3. *The elements to the right of pivots are zero and elements to the left are non-negative and strictly smaller than the pivot.*

The computation of the HNF can be done efficiently in $\mathcal{O}(n^\theta \mathsf{M}(n \log M))$ time and $\mathcal{O}(n^2 \log M)$ space, where $n^\theta$ is the arithmetic complexity of the multiplication of two $n \times n$ matrices and $\mathsf{M}(b) = \mathcal{O}(b)$ the complexity of the multiplication of two $b$-bit integers (see [33] for more details).

### A.2 Ring of integers, integer ideals

**Integers of a number field.** An element $\gamma$ of $\mathbb{K}$ is said to be *integral* if its minimal polynomial has integers coefficients and is monic. The *ring of integers* of $\mathbb{K}$ is the ring of all integral elements contained in $\mathbb{K}$, and is denoted by $\mathcal{O}_{\mathbb{K}}$. Noticeably, the norm of any integer of the number field is an integer.

For $\alpha$ a primitive element of $\mathbb{K}$, we have $\mathbb{Z}[\alpha] \subset \mathcal{O}_{\mathbb{K}}$, but $\mathbb{Z}[\alpha]$ can be strictly included in $\mathcal{O}_{\mathbb{K}}$. Yet, as a finite-rank sub-module of the field $\mathbb{K}$, there exists a finite family $(b_i)_{i\in i}$ such that $\mathcal{O}_{\mathbb{K}} \cong \bigoplus_{i\in I} \mathbb{Z}\cdot\mathbf{b}_i$. Such a family is called an *integral basis* of the number field.

**Ideals and norms.** An additive subgroup $\mathfrak{a}$ of $\mathcal{O}_{\mathbb{K}}$ such that for every $\mathbf{x} \in \mathfrak{a}$ the coset $\mathbf{x}\cdot\mathcal{O}_{\mathbb{K}} = \{\mathbf{x}\cdot\mathbf{a} | \mathbf{a} \in \mathcal{O}_{\mathbb{K}}\}$ lies in $\mathfrak{a}$, is called an *integer ideal* of the number field. One can generalize the notion of norm of an element in the number field to norm of an integer ideal: let define the norm[15] $\mathcal{N}$ as the integer valued map:

$$\mathfrak{a} \mapsto [\mathcal{O}_{\mathbb{K}} : \mathfrak{a}] = \left|\mathcal{O}_{\mathbb{K}}/\mathfrak{a}\right|.$$

The ideal norm is multiplicative: for any ideals $\mathfrak{a}, \mathfrak{b}$: $\mathcal{N}(\mathfrak{a}\cdot\mathfrak{b}) = \mathcal{N}(\mathfrak{a})\cdot\mathcal{N}(\mathfrak{b})$. Moreover this norm is closely linked to the norm of integers in the sense that for every $\mathbf{a} \in \mathcal{O}_{\mathbb{K}}$, $\mathcal{N}(\langle\mathbf{a}\rangle)) = \left|\mathcal{N}_{\mathbb{K}/\mathbb{Q}}(\mathbf{a})\right|$, where $\langle\mathbf{a}\rangle$ denotes the principal ideal generated by $\mathbf{a}$: $\langle\mathbf{a}\rangle = \{\mathbf{a}\cdot\mathbf{x} | \mathbf{x} \in \mathcal{O}_{\mathbb{K}}\}$.

The norm of an ideal $\mathfrak{a}$ can be used to give an upper bound on the norm of the smallest nonzero element it contains: there always exists a nonzero $\mathbf{a} \in \mathfrak{a}$ for which:

$$\left|\mathcal{N}_{\mathbb{K}/\mathbb{Q}}(\mathbf{a})\right| \leq \left(\frac{2}{\pi}\right)^{r_2} \sqrt{|\Delta_{\mathbb{K}}|}\mathcal{N}(\mathfrak{a}),$$

where $\Delta_{\mathbb{K}}$ is the discriminant of $\mathbb{K}$ and $r_2$ is the number of pairs of complex embeddings, defined as previously.

### A.3 Dirichlet Unit Theorem

**Unit group of a number field.** Let $\mathbb{K}$ be a number field. The *unit group* $\mathcal{U}_{\mathbb{K}}$ of $\mathbb{K}$ is the group of all integers in $\mathcal{O}_{\mathbb{K}}$ whose inverse also lies in $\mathcal{O}_{\mathbb{K}}$. The unit group has a simple geometric characterization in term of norm:

**Lemma 6.** *An element $\alpha \in \mathcal{O}_{\mathbb{K}}$ is an unit if and only if $\mathcal{N}_{\mathbb{K}/\mathbb{Q}}(\alpha) = 1$.*

**Log-Unit lattice.** Let $n = [\mathbb{K} : \mathbb{Q}]$ the degree of the number field, written as $n = r_1 + 2r_2$, where $r_1$ and $r_2$ are defined respectively as the number of real and pairs of complex embeddings. Define the map Log by

$$\begin{aligned}\mathrm{Log}(\mathbf{x}) : \quad & \mathbb{K} \longrightarrow \mathbb{R}^{r_1+r_2} \\ & \mathbf{x} \mapsto \big(\log|\sigma_1(\mathbf{x})|, \cdots, \log|\sigma_{r_1}(\mathbf{x})|, 2\log|\sigma_{r_1+1}(\mathbf{x})|, \cdots, 2\log|\sigma_{r_1+r_2}(\mathbf{x})|\big).\end{aligned}$$

[15] We define here the *absolute norm* of an ideal.

The image of the kernel of Log by the canonical embedding $\sigma$ lies in the intersection between the embedding $\sigma(\mathcal{O}_{\mathbb{K}})$ and the set of points of coordinates lower than 1. Since the embedding of $\mathcal{O}_{\mathbb{K}}$ is discrete, we deduce that $\sigma(\text{Ker Log})$ and so Ker Log are discrete.

Moreover the image $\text{Log}(\mathcal{U}_{\mathbb{K}})$ lies in the hyperplane of equation $x_1 + \cdots + x_{r_1} + x_{r_1+1} + \cdots + x_{r_1+r_2} = 0$. A careful analysis of this image shows that it is in fact a full-rank lattice of this hyperplane. This is the *log-unit* lattice associated to $\mathbb{K}$. These remarks on the map Log lead then to the complete description of the structure of $\mathcal{U}_{\mathbb{K}}$.

**Theorem 6 (Dirichlet's Unit Theorem).** *Let $\mathbb{K}$ be a number field of degree $n = r_1 + 2r_2$ with $r_1$ and $r_2$ the number of real and pairs of complex embeddings. Then, the unit group of $\mathbb{K}$ is a direct product of a discrete cyclic group with a free abelian group of rank $r = r_1 + r_2 - 1$.*