REDUCING THE COMPLEXITY FOR CLASS GROUP COMPUTATIONS USING SMALL DEFINING POLYNOMIALS

ALEXANDRE GÉLIN

Abstract. In this paper, we describe an algorithm that efficiently collect relations in class groups of number fields defined by a small defining polynomial. This conditional improvement consists in testing directly the smoothness of principal ideals generated by small algebraic integers. This strategy leads to an algorithm for computing the class group whose complexity is possibly as low as $L_{|\Delta_{\mathbf{K}}|}\left(\frac{1}{3}\right)$.

1. Introduction

The ideal class group of a number field is a finite abelian group and its computation is a major task in algorithmic algebraic number theory. The case of quadratic number fields was firstly addressed by Shanks [Sha69, Sha72]. Thanks to the baby-step–giant-step strategy and under the Generalized Riemann Hypothesis (GRH), he reached an exponential runtime $O(|\Delta_{\mathbf{K}}|^{\frac{1}{5}})$, where $\Delta_{\mathbf{K}}$ denotes the absolute discriminant of the considered number field.

Hafner and McCurley [HM89] then proposed an algorithm in heuristic subexponential time $L_{|\Delta_{\mathbf{K}}|}(\frac{1}{2}, \sqrt{2})$, but only in the restrictive case of imaginary quadratic number fields. This L-notation is classical when presenting index calculus algorithms with subexponential complexity. Given two constants α and c with $\alpha \in [0, 1]$ and $c \geq 0$, $L_N(\alpha, c)$ is used as a shorthand for

$$\exp\left((c + o(1))(\log N)^{\alpha}(\log\log N)^{1-\alpha}\right),$$

where $o(1)$ tends to 0 as N tends to infinity. We also encounter the notation $L_N(\alpha)$ when specifying c is undesired.

An extension of this latter algorithm to all number fields was the topic of Buchmann's work [Buc90], assuming that the extension degree, arbitrary, is fixed. Then he obtained a heuristic runtime $L_{|\Delta_{\mathbf{K}}|}(\frac{1}{2}, 1.7)$. Finally, Biasse and Fieker improved this algorithm and achieved a subexponential complexity for all number fields, without any restriction on the degree: a complexity $L_{|\Delta_{\mathbf{K}}|}(\frac{2}{3} + \varepsilon)$ in the general case[1] and $L_{|\Delta_{\mathbf{K}}|}(\frac{1}{2})$ when the extension degree n satisfies $n \leq (\log|\Delta_{\mathbf{K}}|)^{3/4-\varepsilon}$. Recently, these complexities were reduced to $L_{|\Delta_{\mathbf{K}}|}\left(\frac{2\alpha+1}{5}, o(1)\right)$ for number fields in classes $\mathcal{D}_{n_0,d_0,\alpha,\gamma}$ with $\alpha > 3/4$, and $L_{|\Delta_{\mathbf{K}}|}\left(\frac{1}{2}, \frac{\omega+1}{2\sqrt{\omega}}\right)$ in the other cases — the classes $\mathcal{D}$ are defined in [GJ16] and the complexities come from [Gél18].

In addition, there exists some conditional improvements when the defining polynomial of the number field has good properties — namely small coefficients. Biasse and Fieker [BF14] achieved an $L_{|\Delta_{\mathbf{K}}|}(a)$ complexity with a possibly as low as $\frac{1}{3}$, and this improvement has been widened in [GJ16] to a larger set of number fields.

[1] For an arbitrary small $\varepsilon > 0$.

Contribution. In this paper, we focus on a conditional improvement based on the smallness of the defining polynomial. Though ideal-reduction schemes enforce an $L_{|\Delta_{\mathbf{K}}|}\left(\frac{1}{2}\right)$ complexity, the solution of the discrete logarithm problem in finite fields in $L_Q\left(\frac{1}{3}\right)$ suggests that we can reach this value for class group computations too. This is the aim of the *sieving strategy*. We first describe the algorithm and extend the results obtained by Biasse in [Bia14]. Then we study its complexity, compare it with the results of [Gél18] and exhibit the number fields for which this new strategy offers a better complexity than ideal reductions. In addition, we provide an algorithm for solving the Principal Ideal Problem by using techniques close to the ones used for class group computations.

Outline. The article is organized as follows. In Section 2 we briefly explain how this sieving strategy may speed up class group computation. Then Section 3 is devoted to the description of the relation collection algorithm, while Section 4 gives the parameter choices together with the complexity analysis according to the classes $\mathcal{D}$. Section 5 summarizes where each algorithm — this one and the one based on ideal reduction — is better than the other in order to give a new state of the art of class group computation. Finally, the solution of the Principal Ideal Problem based on this method is provided in Section 6.

2. Motivation

As it is explained in [Gél18, Section 2], computing class groups and regulators in number fields is essentially based on the index calculus method. Within this strategy, the part that determines the complexity is the relation collection, because the linear-algebra step only leads to an additional constant factor in the exponent — *i.e.,* in the second constant in the L-notation. The relation collection step, as its name suggests, consists in searching for many principal ideals that split over the factor base $\mathcal{B} = \{\mathfrak{p}_1, \ldots, \mathfrak{p}_N\}$ composed of all prime ideals of norm below a bound $B > 0$:

$$\langle x \rangle \, \boldsymbol{\mathcal{O}}_{\mathbf{K}} = \prod \mathfrak{p}_i^{e_i} \qquad \text{for } x \in \boldsymbol{\mathcal{O}}_{\mathbf{K}}.$$

In the general case, without making any assumption on the number fields, the ideal-reduction strategy performs best and leads to a complexity that is at least $L_{|\Delta_{\mathbf{K}}|}\left(\frac{1}{2}\right)$. However, there exist conditional improvements when the number field is defined by a *good* polynomial, that is a polynomial having small height. Indeed, in that case, the $\mathfrak{q}$-descent strategy described by Biasse and Fieker in [BF14] and generalized in [GJ16] allows a complexity between $L_{|\Delta_{\mathbf{K}}|}\left(\frac{1}{3}\right)$ and $L_{|\Delta_{\mathbf{K}}|}\left(\frac{1}{2}\right)$ for all number fields of small extension degree.

Our new idea that underlies this article is to generate the relations by testing a lot of *small* principal ideals that are generated by algebraic integers of bounded degree and coefficients. The norms of such elements depend on the two bounds used for the degree and on the coefficients and the height of the defining polynomial. This idea was already used in the Number Field Sieve [LLMP90]. Enge, Gaudry, and Thomé [EG07, EGT11] extend this method to low-degree curves for solving the discrete logarithm problem over such curves in $L_{q^g}\left(\frac{1}{3}\right)$, where q is the cardinality of the base field and g the genus of the curve.

Then, Biasse in [Bia14] applies the method in the context of class group computations. His result only addresses very specific number fields $\mathbf{K}$ defined by a

polynomial T such that

$$\big[\mathbf{K} : \mathbf{Q}\big] \leq O(\log|\Delta_{\mathbf{K}}|)^{\alpha} \quad \text{and} \quad \log H(T) \leq O(\log|\Delta_{\mathbf{K}}|)^{1-\alpha} \tag{1}$$

for an α in the open interval $\left(\frac{1}{3}, \frac{2}{3}\right)$. In so doing, he was able to compute the class group in time $L_{|\Delta_{\mathbf{K}}|}\left(\frac{1}{3}\right)$ assuming the Extended Riemann Hypothesis (ERH) and under heuristics. We generalize here the sieving strategy to all number fields, obtaining a complexity possibly as low as $L_{|\Delta_{\mathbf{K}}|}\left(\frac{1}{3}\right)$.

This method has also been used by Buchmann, Jacobson, Neis, Theobald, and Weber in [BJN+99] for practical enhancements. Indeed, the sieving strategy definitely outperforms ideal reduction in practice, especially for small-degree number fields.

The $\mathfrak{q}$-descent strategy explained in [BF14], where elements with small coefficients are searched in lattices of smaller dimension, is, in a certain sense, another way to use these small algebraic integers. However, our method appears easier to understand and its complexity analysis is streamlined: we are able to provide explicitly the second constant in the L-notation, which does not sound that simple for the $\mathfrak{q}$-descent. In addition, from a practical point of view, as the $\mathfrak{q}$-descent only works in small degree $\left(\alpha \leq \frac{1}{2}\right)$, our algorithm should outperform the $\mathfrak{q}$-descent, since it does not require iterations nor lattice-reductions.

3. Deriving relations by sieving

In the following, we make use of the classification presented in [Gél18] based on the classes $\mathcal{D}$ introduced in [GJ16]:

Definition 3.1 ([Gél18, Definition 3.1])**.** Let $n_0 > 1$ be a real parameter arbitrarily close to 1, $d_0 > 0$, $\alpha \in [0, 1]$ and $\gamma \geq 1 - \alpha$. The class $\mathcal{D}_{n_0,d_0,\alpha,\gamma}$ is defined as the set of all number fields $\mathbf{K}$ of discriminant $\Delta_{\mathbf{K}}$ that admit a monic defining polynomial $T \in \mathbf{Z}[X]$ of degree n that satisfies:

$$\begin{aligned} \frac{1}{n_0}\left(\frac{\log|\Delta_{\mathbf{K}}|}{\log\log|\Delta_{\mathbf{K}}|}\right)^{\alpha} \quad &\leq \quad n \quad \leq \quad n_0\left(\frac{\log|\Delta_{\mathbf{K}}|}{\log\log|\Delta_{\mathbf{K}}|}\right)^{\alpha} \qquad \text{and} \\ d = \log H(T) \quad &\leq \quad d_0(\log|\Delta_{\mathbf{K}}|)^{\gamma}(\log\log|\Delta_{\mathbf{K}}|)^{1-\gamma}. \end{aligned} \tag{2}$$

For a fixed number field $\mathbf{K}$ in a class $\mathcal{D}_{n_0,d_0,\alpha,\gamma}$, the value $\alpha \in [0, 1]$ corresponds to the extension degree so that it is precisely defined. For the second main parameter $\gamma \geq 1 - \alpha$, special care should be taken: sometimes it costs too much to reduce the defining polynomial. This issue is addressed in Section 5: given a number field defined by a polynomial, we study the optimal strategy for computing the class group depending on the parameters. Is the polynomial reduction necessary? Is it better to use ideal-reduction or sieving?

Remark 3.2. We use the terminology *"sieving strategy"* because it closely corresponds to the way to — efficiently — implement it. Theoretically, our algorithm only consists in testing for smoothness a huge arithmetic progression of algebraic integers until we have found sufficiently many relations.

The description of the algorithm we are going to introduce is clear and the algorithm is easily understandable. Difficulties arise when we need to fix the parameters such as the smoothness bound for the factor base and the bounds that describe the sieving space in order to minimize the complexity. To fix the notation, we consider

a number field $\mathbf{K} = \mathbf{Q}(\theta)$ of degree n and let T denote the defining polynomial of which θ is a root.

Let $B > 0$ be the smoothness bound that must be determined. We fix the factor base $\mathcal{B} = \{\mathfrak{p}_1, \dots, \mathfrak{p}_N\}$ as the set of all prime ideals of $\boldsymbol{\mathcal{O}}_{\mathbf{K}}$ whose norm is below B. From the Landau Prime Ideal Theorem [Lan03], we know that its cardinality satisfies

$$N = |\mathcal{B}| = B\big(1 + o(1)\big).$$

We describe the sieving space by fixing a bound $t > 0$ on the degree, together with a bound $S > 0$ on the coefficients. Hence we use all the polynomials of degree at most t with coefficients between $-S$ and S. These are $(2S+1)^{t+1}$ polynomials, but only half of them are of interest, as algebraic integers x and $-x$ generate the same ideal. Note that we may also avoid algebraic integers built from a reducible polynomial in θ. Indeed, if $x = x_1 \cdot x_2$, then the exponents of a relation produced by x equal the sums of the exponents of relations produced by x_1 and x_2.

Given an algebraic integer $x = \sum_{i=0}^{t} a_i \theta^i$ and denoting by A the polynomial $A(X) = \sum a_i X^i$, the norm of the principal ideal $\langle x \rangle$ is given by

$$\mathcal{N}\big(\langle x \rangle\big) = \mathcal{N}_{\mathbf{K}/\mathbf{Q}}\big(x\big) = \mathrm{Res}\,(A, T)\,.$$

The bounds for the resultants displayed in [BL10, Theorem 7] allow us to provide a bound on the field norm of an element given in standard representation. Thus, thanks to the two bounds t on the degree and S on the coefficients, we can derive an upper bound for the norm of the principal ideal $\langle x \rangle$:

$$\mathcal{N}\big(\langle x \rangle\big) \leq \sqrt{t+1}^{\,n} \sqrt{n+1}^{\,t} H(T)^t S^n. \tag{3}$$

We also recalled the two heuristics used in [Gél18], as we also need them.

Heuristic 3.3 ([Gél18, Heuristic 4.4])**.** *The probability $\mathcal{P}(x, y)$ that an ideal of norm bounded by x is y-smooth satisfies*

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

Heuristic 3.4 ([Gél18, Heuristic 4.7])**.** *There exists K negligible compared with $|\mathcal{B}|$ such that collecting $K \cdot |\mathcal{B}|$ relations suffices to obtain a relation matrix that generates the whole lattice of relations.*

Assuming Heuristic 3.3, the previous bound on the norm offers a lower bound on the probability $\mathcal{P}$ of B-smoothness of any principal ideal $\langle x \rangle$ belonging to the sieving space. Then the $(2S+1)^{t+1}$ small ideals lead to $(2S+1)^{t+1} \cdot \mathcal{P}$ relations. Assuming Heuristic 3.4, collecting $N\big(1 + o(1)\big)$ relations suffices to derive the class group. Therefore we want the following relation to be satisfied by our choice of parameters:

$$(2S+1)^{t+1} \cdot \mathcal{P} = N\big(1 + o(1)\big). \tag{4}$$

Remark 3.5. Note that making use of the weaker Heuristic 3.4, introduced in [GJ16], is essential here. Indeed, the factor base may contain ideals of degree $k > t$, that cannot be part of any relations derived from our settings. Because every ideal whose norm is below the Bach bound has a degree smaller than $\log 12 + 2 \log\log |\Delta_{\mathbf{K}}|$, we know that sieving on degree-t polynomials suffices for our purposes, which was not the case with the heuristic used before, where the relation matrix must have full rank.

To evaluate the cost of the sieving phase, we need to know the number of ideals we test for smoothness: it is $(2S+1)^{t+1}$. We explain below that the cost of each smoothness test is always negligible. Then the overall cost of the sieving phase is given by $(2S+1)^{t+1}(1+o(1))$.

As the lowest final complexity is obtained when a balance is reached between the cost of the relation collection and the cost of the linear-algebra phase, we also want that

$$(2S+1)^{t+1} = N^{\omega+1}\big(1+o(1)\big), \tag{5}$$

because the linear algebra cost is in $N^{\omega+1}$ (see [BF14, Proposition 4.1]).

Before determining the parameters that minimize the complexity, we give an outline of the strategy in Algorithm 1.

Algorithm 1 Deriving relations from small algebraic integers

Input: The factor base $\mathcal{B}$, the degree bound t and the coefficient bound S.
Output: The relations stored.

for d **from** 1 **to** t **do**
  for all $(a_0, \ldots, a_d) \in [-S, \ldots, S]^{d+1}$ **do**
    Fix $x = \sum a_i \theta^i$ and $\mathfrak{a} = \langle x \rangle$
    Test the B-smoothness of $\mathfrak{a}$
    if $\mathfrak{a}$ is B-smooth **then**
      Fix e_i such that $\mathfrak{a} = \prod \mathfrak{p}_i^{e_i}$
      Store the relation $\langle x \rangle = \prod \mathfrak{p}_i^{e_i}$
    end if
  end for
end for

We describe in the subsequent sections how to set the parameters for the factor base and the sieving space to achieve the best complexities. We fix $n_0 > 1$, $d_0 > 0$, $\alpha \in [0,1]$ and $\gamma \geq 1-\alpha$ and let $\mathbf{K}$ be a number field that belongs to $\mathcal{D}_{n_0,d_0,\alpha,\gamma}$. We also assume that we know a *good* defining polynomial T that satisfies

$$\log H(T) \quad \leq \quad d_0 (\log |\Delta_{\mathbf{K}}|)^{\gamma} (\log\log |\Delta_{\mathbf{K}}|)^{1-\gamma}.$$

Let θ be a primitive element of $\mathbf{K}$ that is a root of the defining polynomial T. As in the discrete logarithm problem in finite fields, we need to distinguish several cases according to the relative sizes of α and γ. However, the distinctions between the various cases are not as precise as they are for the DLP: we consider small, medium and large degrees and give the corresponding inequalities involving α and γ.

4. Complexity analyses

4.1. **The case of medium degree.** We begin by the medium case, which we define by α and γ being of the same magnitude. This includes $\alpha \approx \gamma \approx \frac{1}{2}$, but covers a much wider range as follows from the analysis below. As already discussed at the beginning of [Gél18, Section 3], the size of the defining polynomial plays a role in the complexity: we only have the inequality $\gamma \geq 1-\alpha$, so that we have no choice but to keep using both α and γ.

Given that we hope to find an algorithm with runtime $L_{|\Delta_{\mathbf{K}}|}\left(\frac{1}{3}\right)$ and given that $\gamma \geq 1-\alpha$ (thus $\alpha+\gamma \geq 1$), we simply conjecture the existence of an algorithm

with runtime $L_{|\Delta_{\mathbf{K}}|}\left(\frac{\alpha+\gamma}{3}\right)$ and fix the size of the factor base $\mathcal{B}$ as the set of prime ideals of norm at most

$$B = L_{|\Delta_{\mathbf{K}}|}\left(\frac{\alpha+\gamma}{3}, c_b\right),$$

with $c_b > 0$ to be determined. The notation L is identical as the L introduced earlier, except that we have removed the $o(1)$, in order to consider constants: $L_N(\alpha, c) = e^{c(\log N)^\alpha (\log\log N)^{1-\alpha}}$.

Thanks to Landau's Prime Ideal Theorem [Lan03], we know that $N = |\mathcal{B}| = L_{|\Delta_{\mathbf{K}}|}\left(\frac{\alpha+\gamma}{3}, c_b\right)$. The sieving space is chosen to consist in all algebraic integers $x = A(\theta)$, built as polynomials in θ, that satisfy

(6)

$$\deg A \leq t = c_t\left(\frac{\log|\Delta_{\mathbf{K}}|}{\log\log|\Delta_{\mathbf{K}}|}\right)^{\frac{2}{3}(\alpha+\gamma)-\gamma} \text{ and } H(A) \leq S = L_{|\Delta_{\mathbf{K}}|}\left(\frac{2}{3}(\alpha+\gamma)-\alpha, c_s\right).$$

In particular, $\log H(A) \leq c_s(\log|\Delta_{\mathbf{K}}|)^{\frac{2}{3}(\alpha+\gamma)-\alpha}(\log\log|\Delta_{\mathbf{K}}|)^{1-\left(\frac{2}{3}(\alpha+\gamma)-\alpha\right)}$. So these two quantities are only well defined for $\frac{2}{3}(\alpha+\gamma)-\gamma \geq 0$ and $\frac{2}{3}(\alpha+\gamma)-\alpha \geq 0$, which defines the bounds of the medium-degree case.

According to Equation (3), this choice of parameters enables to bound the norm of every principal ideal $\langle x\rangle$ in the sieving space by

$$\mathcal{N}\big(\langle x\rangle\big) \leq L_{|\Delta_{\mathbf{K}}|}\left(\frac{2}{3}(\alpha+\gamma), n_0c_s + d_0c_t\right). \tag{7}$$

We deduce from Heuristic 3.3 that a principal ideal generated by such an x is B-smooth with probability

$$\mathcal{P} \geq L_{|\Delta_{\mathbf{K}}|}\left(\frac{\alpha+\gamma}{3}, \frac{(\alpha+\gamma)(n_0c_s+d_0c_t)}{3c_b}\right)^{-1}.$$

The size of the sieving space is given by $(2S+1)^{t+1} = L_{|\Delta_{\mathbf{K}}|}\left(\frac{\alpha+\gamma}{3}, c_sc_t\right)$. As usual, this estimation allows us to estimate the number of relations found by combining the two previous results: the number of collected relations is expected to be

$$(2S+1)^{t+1}\cdot\mathcal{P} = L_{|\Delta_{\mathbf{K}}|}\left(\frac{\alpha+\gamma}{3}, c_sc_t - \frac{(\alpha+\gamma)(n_0c_s+d_0c_t)}{3c_b}\right).$$

With $N = L_{|\Delta_{\mathbf{K}}|}\left(\frac{\alpha+\gamma}{3}, c_b\right)$ and the assumption of Heuristic 3.4 (see Equation (4)), we obtain

$$c_sc_t - \frac{(\alpha+\gamma)(n_0c_s+d_0c_t)}{3c_b} = c_b.$$

Another equation between the various constants stems from the balance between the relation collection and the linear algebra, as stated by Equation (5). It boils down to

$$c_sc_t = (\omega+1)c_b.$$

From these two equations, we easily express c_t in the other constants and obtain a deg-2 equation in c_b, depending on c_s: $3\omega c_sc_b^2 - d_0(\alpha+\gamma)(\omega+1)c_b - n_0(\alpha+\gamma)c_s^2 = 0$. This expression allows us to infer the shape of c_b, which is going to give us the final complexity, depending on c_s:

$$c_b = \frac{d_0(\alpha+\gamma)(\omega+1) + \sqrt{d_0^2(\alpha+\gamma)^2(\omega+1)^2 + 12n_0(\alpha+\gamma)\omega c_s^3}}{6\omega c_s}.$$

It only remains to minimize this quantity as a function of c_s. It follows from a straight analysis that the minimum is achieved for c_s satisfying $c_s^3 = \frac{2d_0^2(\alpha+\gamma)(\omega+1)^2}{3n_0\omega}$, which leads to

$$c_b = \left(\frac{4n_0 d_0(\alpha+\gamma)^2(\omega+1)}{9\omega^2}\right)^{\frac{1}{3}}.$$

Consequently, the runtime of our algorithm for computing the class group structure and an approximation of the regulator is

$$L_{|\Delta_{\mathbf{K}}|}\left(\frac{\alpha+\gamma}{3}, \left(\frac{4n_0 d_0(\alpha+\gamma)^2(\omega+1)^4}{9\omega^2}\right)^{\frac{1}{3}}\right).$$

Remark 4.1. The first constant may be as low as $\frac{1}{3}$ if γ reaches the lower bound $1-\alpha$, *i.e.,* $\alpha+\gamma = 1$.

We also mention that in this case, our second constant is better than the one found by Biasse in [Bia14].

This analysis however only holds when the two quantities $\frac{2}{3}(\alpha+\gamma) - \gamma$ and $\frac{2}{3}(\alpha+\gamma) - \alpha$ are non-negative. These conditions offer the limits of our analysis and can be rewritten as

$$\frac{1}{3}(\alpha+\gamma) \leq \alpha \leq \frac{2}{3}(\alpha+\gamma) \qquad \Longleftrightarrow \qquad \frac{\gamma}{2} \leq \alpha \leq 2\gamma.$$

Therefore, it remains to treat the two complementary cases, when either the size of the defining-polynomial height or the extension degree prevails.

4.2. **The small-degree case: when $2\alpha < \gamma$.** The first extreme case we study is when the size of the defining-polynomial height outweighs the extension degree. It corresponds to the left part of the diagrams displayed in [Gél18], where the **q**-descent strategy works. In these cases, the extension degree satisfies

$$\alpha < \frac{\gamma}{2} \qquad \Longleftrightarrow \qquad \alpha < \frac{1}{3}(\alpha+\gamma).$$

We are able to reach a final complexity in $L_{|\Delta_{\mathbf{K}}|}\left(\frac{\gamma}{2}\right)$ for the relation collection. As α is relatively small — below $\frac{\gamma}{2}$ — we know that the defining-polynomial reduction algorithm presented in [GJ16] runs in time $L_{|\Delta_{\mathbf{K}}|}(\alpha)$, which is strictly less than $L_{|\Delta_{\mathbf{K}}|}\left(\frac{\gamma}{2}\right)$. Hence this reduction is always negligible compared with the relation collection, so that it can be considered as a precomputation. According to [GJ16, Corollary 3.3], we can also assume $\gamma \leq 1$.

We fix the size of the factor base $\mathcal{B}$ by considering all the prime ideals having norm below

$$B = L_{|\Delta_{\mathbf{K}}|}\left(\frac{\gamma}{2}, c_b\right),$$

and we have from Landau's theorem that $N = |\mathcal{B}| = L_{|\Delta_{\mathbf{K}}|}\left(\frac{\gamma}{2}, c_b\right)$. The sieving space is constructed as before, using all polynomials A that satisfy

$$\deg A \leq t = c_t \quad \text{and} \quad H(A) \leq S = L_{|\Delta_{\mathbf{K}}|}\left(\frac{\gamma}{2}, c_s\right). \tag{8}$$

These adjustments in the definition are motivated by the desire to minimize the norm size. As the height of the defining polynomial is large, we bound the degree of the algebraic integers to guarantee that the norm stays small.

According to Equation (3), this choice of parameters enables to bound the norm of every principal ideal $\langle x \rangle$ in the sieving space by

$$\mathcal{N}\big(\langle x \rangle\big) \leq L_{|\Delta_{\mathbf{K}}|}(\gamma, d_0 c_t). \tag{9}$$

Assuming Heuristic 3.3 allows us to have the following inequality satisfied by the probability for a principal ideal generated by such an x to be B-smooth:

$$\mathcal{P} \geq L_{|\Delta_{\mathbf{K}}|}\left(\frac{\gamma}{2}, \frac{d_0 \gamma c_t}{2c_b}\right)^{-1}.$$

As the sieving-space cardinality is $(2S+1)^{t+1} = L_{|\Delta_{\mathbf{K}}|}\left(\frac{\gamma}{2}, c_s(c_t+1)\right)$, we obtain the number of collected relations as before and Equation (4) results in $c_s(c_t+1) - \frac{d_0\gamma c_t}{2c_b} = c_b$. Similarly Equation (5) leads to $c_s(c_t+1) = (\omega+1)c_b$. From an identical approach as in the previous section, we find the optimal choices for the constants and conclude that the runtime of our algorithm is

$$L_{|\Delta_{\mathbf{K}}|}\left(\frac{\gamma}{2}, \left(\frac{d_0\gamma(\omega+1)^2 c_t}{2\omega}\right)^{\frac{1}{2}}\right).$$

Remark 4.2. The first constant is always between $\frac{1}{3}$ and $\frac{1}{2}$: the upper bound is a consequence of the precomputation made for finding the minimal-height defining polynomial while the lower one comes from $\gamma > \frac{2}{3}(\alpha+\gamma) \geq \frac{2}{3}$. In the second constant, the factor c_t appears so that the complexity depends on the degree of the polynomials we use for sieving. The minimal value is obtained for $c_t = 1$, for a runtime in $L_{|\Delta_{\mathbf{K}}|}\left(\frac{\gamma}{2}, \left(\frac{d_0\gamma(\omega+1)^2}{2\omega}\right)^{\frac{1}{2}}\right)$.

Remark 4.3. A possible alternative for the sieving may be to enlarge the sieving space by allowing larger coefficients — always below $S' = L_{|\Delta_{\mathbf{K}}|}(\gamma - \alpha, o(1))$ — and to consider only a random subset of size $L_{|\Delta_{\mathbf{K}}|}\left(\frac{\gamma}{2}, c_s(c_t+1)\right)$ of the sieving space. Using the bound S' does not affect Equation (9) and the complexity is preserved.

4.3. **The large-degree case: when $\alpha > 2\gamma$.** In this last case, the extension degree outweighs the size of the defining-polynomial height. It corresponds to the right part of the diagrams displayed in [Gél18]. Here we have to work with the input defining polynomial because finding the minimal one costs too much. As the extension degree is large, we opt for sieving polynomials that have small coefficients and large degrees.

We fix the size of the factor base $\mathcal{B}$ by considering all the prime ideals having norm below

$$B = L_{|\Delta_{\mathbf{K}}|}\left(\frac{\alpha}{2}, c_b\right),$$

and we have from Landau's theorem that $N = |\mathcal{B}| = L_{|\Delta_{\mathbf{K}}|}\left(\frac{\alpha}{2}, c_b\right)$. The sieving space is constructed using all polynomials A that satisfy

$$\deg A \leq t = c_t\left(\frac{\log|\Delta_{\mathbf{K}}|}{\log\log|\Delta_{\mathbf{K}}|}\right)^{\frac{\alpha}{2}} \quad \text{and} \quad H(A) \leq S = L_{|\Delta_{\mathbf{K}}|}(0, c_s) = (\log|\Delta_{\mathbf{K}}|)^{c_s}. \tag{10}$$

According to Equation (3), this choice of parameters enables to bound the norm of every principal ideal $\langle x \rangle$ in the sieving space by

$$\mathcal{N}\big(\langle x \rangle\big) \leq L_{|\Delta_{\mathbf{K}}|}\left(\alpha, n_0\left(c_s + \frac{\alpha}{4}\right)\right). \tag{11}$$

We deduce from Equation (11) and Heuristic 3.3 that the probability for a principal ideal generated by such an x to be B-smooth satisfies

$$\mathcal{P} \geq L_{|\Delta_{\mathbf{K}}|}\left(\frac{\alpha}{2}, \frac{n_0\alpha(\alpha+4c_s)}{8c_b}\right)^{-1}.$$

Finally, an identical analysis enables to find the optimal choice for the constants. The final runtime for our class group algorithm based on sieving strategy satisfies

$$L_{|\Delta_{\mathbf{K}}|}\left(\frac{\alpha}{2}, \left(\frac{n_0\alpha(\alpha+4c_s)(\omega+1)^2}{8\omega}\right)^{\frac{1}{2}}\right).$$

Remark 4.4. The first constant is always between $\frac{1}{3}$ and $\frac{1}{2}$ since $\alpha > \frac{2}{3}(\alpha+\gamma) \geq \frac{2}{3}$. In the second constant, the constant c_s appears which can be chosen arbitrarily small. The minimal runtime thus becomes $L_{|\Delta_{\mathbf{K}}|}\left(\frac{\alpha}{2}, \left(\frac{n_0\alpha^2(\omega+1)^2}{8\omega}\right)^{\frac{1}{2}}\right)$.

Remark 4.5. Again, it is possible to enlarge the sieving space by allowing the degree to be larger — always below $t' = c_t\left(\frac{\log|\Delta_{\mathbf{K}}|}{\log\log|\Delta_{\mathbf{K}}|}\right)^{\alpha-\gamma-\varepsilon}$ for $\varepsilon > 0$ arbitrarily small — and to consider only a random subset of the sieving space of size $L_{|\Delta_{\mathbf{K}}|}\left(\frac{\alpha}{2}, c_s c_t\right)$. Using the bound t' does not affect Equation (11) and the complexity is preserved.

5. Conclusion on sieving strategy

The complexity analyses we have derived in the previous sections assume that we know a small defining polynomial T, that is a witness to the fact that $\mathbf{K}$ belongs to the class $\mathcal{D}$. We recall that the classes $\mathcal{D}$ satisfy

$$\mathcal{D}_{n_0,d_F,\alpha,\gamma_F} \quad \subset \quad \mathcal{D}_{n_0,d_0,\alpha,\gamma_0},$$

for $n_0, d_0, d_F > 0$, $0 \leq \alpha \leq 1$ and $1-\alpha \leq \gamma_F < \gamma_0$. To identify the best strategy depending on the inputs, we consider a number field $\mathbf{K}$ defined by a polynomial T such that

$$\frac{1}{n_0}\left(\frac{\log|\Delta_{\mathbf{K}}|}{\log\log|\Delta_{\mathbf{K}}|}\right)^{\alpha} \leq \deg T \leq n_0\left(\frac{\log|\Delta_{\mathbf{K}}|}{\log\log|\Delta_{\mathbf{K}}|}\right)^{\alpha} \text{ and}$$
$$\log H(T) \leq d_0(\log|\Delta_{\mathbf{K}}|)^{\gamma_0}(\log\log|\Delta_{\mathbf{K}}|)^{1-\gamma_0}.$$

It is easily verified that $\mathbf{K}$ belongs to $\mathcal{D}_{n_0,d_0,\alpha,\gamma_0}$. In addition we introduce γ_F and d_F so that γ_F is the minimal γ such that $\mathbf{K} \in \mathcal{D}_{n_0,d_F,\alpha,\gamma}$. Thus we consider two different classes to which $\mathbf{K}$ belongs, namely $\mathcal{D}_{n_0,d_F,\alpha,\gamma_F}$ and $\mathcal{D}_{n_0,d_0,\alpha,\gamma_0}$; note that

$$\mathbf{K} \in \mathcal{D}_{n_0,d_F,\alpha,\gamma_F} \subset \mathcal{D}_{n_0,d_0,\alpha,\gamma_0}.$$

Given the number field $\mathbf{K}$ defined by the polynomial T as inputs, we study the different options for computing the class group and give the optimal strategy. Let us first look at the medium-degree case, where $\frac{\gamma_0}{2} \leq \alpha \leq 2\gamma_0$. Necessarily, we have $\alpha \geq \frac{\alpha+\gamma_0}{3} \geq \frac{1}{3}$.

- When $\alpha \leq \frac{1}{2}$, as $\gamma_0 \leq 2\alpha$, we have $\frac{\alpha+\gamma_0}{3} \leq \frac{1}{2}$ and sieving is the best strategy.
- When $\frac{1}{2} < \alpha \leq \frac{3}{4}$, the sieving strategy remains optimal as long as $\frac{\alpha+\gamma_0}{3} \leq \frac{1}{2}$. Indeed, beyond this bound, the ideal-reduction strategy becomes less costly and should be preferred. This happens as soon as $\gamma_0 \geq 1$.

- Similarly, for $\frac{3}{4} < \alpha \leq 1$, the sieving strategy remains optimal as long as $\frac{\alpha+\gamma_0}{3} \leq \frac{2\alpha+1}{5}$. Above this bound, the ideal-reduction strategy becomes the best option. This happens as soon as $\gamma_0 \geq \frac{4}{5}$.

The large-degree case is easier to deal with. Provided that $\alpha > 2\gamma_0$, we know that the sieving strategy results in an algorithm with runtime $L_{|\Delta_\mathbf{K}|}\left(\frac{\alpha}{2}\right)$, between $L_{|\Delta_\mathbf{K}|}\left(\frac{1}{3}\right)$ and $L_{|\Delta_\mathbf{K}|}\left(\frac{1}{2}\right)$, as $\alpha > 2\gamma_0$ implies that $\alpha \geq \frac{2(\alpha+\gamma_0)}{3} \geq \frac{2}{3}$. This is always the best option.

The small-degree case is when defining-polynomial reduction plays a role. Indeed, we know that its cost is $L_{|\Delta_\mathbf{K}|}(\alpha)$ while the sieving strategy runs in time $L_{|\Delta_\mathbf{K}|}\left(\frac{\gamma}{2}\right)$. Because $\alpha < \frac{\gamma_0}{2}$, we can always perform this reduction as a precomputation. It allows to find the smallest-height defining polynomial and so the minimal γ_F. This reduction has two outcomes:

- If $\frac{\gamma_F}{2} < \alpha$, then the sieving strategy has a complexity in $L_{|\Delta_\mathbf{K}|}\left(\frac{\alpha+\gamma_F}{3}\right)$, which is negligible compared to the cost of the reduction, so that the final runtime is $L_{|\Delta_\mathbf{K}|}(\alpha)$. This can only happens when $\alpha > \frac{1}{3}$, since $\alpha+\gamma_F \geq 1$.
- If $\frac{\gamma_F}{2} > \alpha$, then the sieving strategy has a complexity that outweighs the cost of the reduction, so that the final runtime is $L_{|\Delta_\mathbf{K}|}\left(\frac{\gamma_F}{2}\right)$. This value is between $L_{|\Delta_\mathbf{K}|}\left(\frac{1}{3}\right)$ and $L_{|\Delta_\mathbf{K}|}\left(\frac{1}{2}\right)$, as the reduction algorithm returns a polynomial such that $\gamma_F \leq 1$ — this is a direct consequence of [GJ16, Corollary 3.3] — and because $\gamma_F > 2\alpha$ implies that $\gamma_F \geq \frac{2(\alpha+\gamma_0)}{3} \geq \frac{2}{3}$. This is the only option when $\alpha < \frac{1}{3}$.

The results of this analysis are summarized in Table 1. We also give a new diagram for the complexities in Figure 1.

Cond. on α	**Cond. on γ**	**Strategy**	**Complexity**
$\alpha \leq \frac{1}{2}$	$\gamma_0 \leq 2\alpha$	Sieving (MD)	$L_{\lvert\Delta_\mathbf{K}\rvert}\left(\frac{\alpha+\gamma_0}{3}\right)$
	$2\alpha < \gamma_F \leq \gamma_0$	Pol. Red. & Sieving (SD)	$L_{\lvert\Delta_\mathbf{K}\rvert}\left(\frac{\gamma_F}{2}\right)$
	$\gamma_F < 2\alpha < \gamma_0$	Pol. Red. & Sieving (SD)	$L_{\lvert\Delta_\mathbf{K}\rvert}(\alpha)$
$\alpha > \frac{1}{2}$	$2\gamma_0 \leq \alpha$	Sieving (LD)	$L_{\lvert\Delta_\mathbf{K}\rvert}\left(\frac{\alpha}{2}\right)$
	$\frac{\alpha+\gamma_0}{3} \leq \max\left(\frac{1}{2}, \frac{2\alpha+1}{5}\right)$	Sieving (MD)	$L_{\lvert\Delta_\mathbf{K}\rvert}\left(\frac{\alpha+\gamma_0}{3}\right)$
	$\frac{\alpha+\gamma_0}{3} > \max\left(\frac{1}{2}, \frac{2\alpha+1}{5}\right)$	Ideal Reduction	$L_{\lvert\Delta_\mathbf{K}\rvert}\left(\max\left(\frac{1}{2}, \frac{2\alpha+1}{5}\right)\right)$

TABLE 1. Choice of the strategy depending on the input parameters.

6. Application to Principal Ideal Problem

In addition to the step forward for class group computations, our results allow us to improve the resolution of another problem: the Principal Ideal Problem (PIP). It consists in finding a generator of an ideal, assuming it is principal. The Short Principal Ideal Problem (SPIP) follows from the PIP by adding the assumption that there exists a *small* generator. The SPIP is the base of several Fully Homomorphic Encryption schemes inspired by the work of Gentry [Gen09] such as the FHE scheme presented by Smart and Vercauteren at PKC 2010 [SV10] and the multilinear map scheme presented by Garg, Gentry, and Halevi at EuroCrypt in 2013 [GGH13]. Solving the SPIP is a two-stage process that consists of first solving the underlying

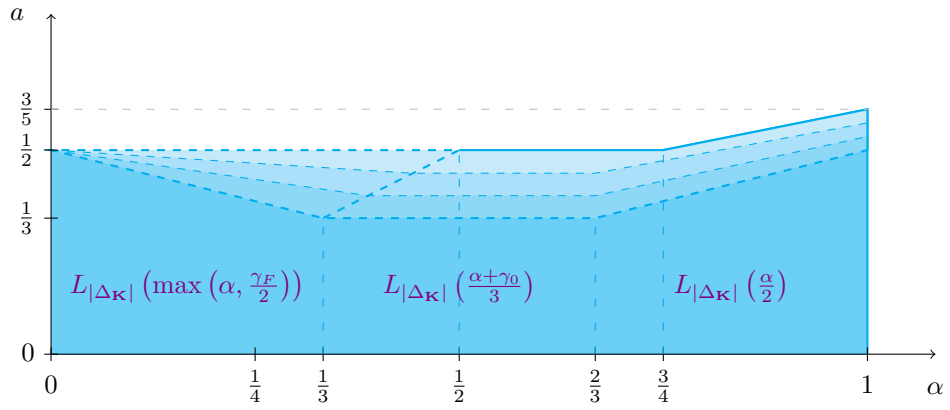

FIGURE 1. Complexity obtained with our sieving strategy.

PIP (on which we focus here), if successful followed by attempts to reduce the generator found to a short one (see [CDPR16] for instance). Finding a generator of a principal ideal, and even testing the principality of an ideal, are difficult problems in algorithmic number theory, as described in detail in [Coh93, Chapter 4] and [Thi95, Section 7].

The general strategy is similar to the one used for the Discrete Logarithm Problem in finite fields. Indeed, for finding the logarithm of an element, two steps are distinguished: first, we find the logarithms of many small elements; second, we express our target element using these small elements and recover its logarithm. It is the same here with our ideal $\mathfrak{a}$, assumed to be principal. First, we compute the matrix of relations as for class group computations, keeping track of the small elements we have sieved with. Second, we find an ideal $\mathfrak{b}$ that is in the same class as $\mathfrak{a}$ and that splits over the factor base. Then, linear algebra allows us to recover a generator of $\mathfrak{b}$ thanks to the relation matrix and finally, we can solve the PIP.

6.1. **The descent algorithm.** We first briefly outline the algorithm without fixing the parameters. Indeed, as for class group computations, the optimal parameters choices are derived from the complexity analyses, depending on the number-field exponents α and γ. In order to bootstrap the descent, we start with a classical BKZ-reduction to obtain an ideal of reasonable norm. Indeed, as the input ideal $\mathfrak{a}$ is fixed — the one for which we want a generator — it can have an arbitrarily large norm. All the ideal reductions are performed on the lattice built from the coefficient embedding $\varsigma(\mathfrak{a})$, as it is described in [BEF+17, Section 2.2]. The block-size is fixed so that the complexity of the reduction is strictly below the overall complexity of the algorithm, as it is done in [Gél18, Section 5]. Then the descent consists in a succession of ideal reductions and smoothness tests so that the norms of all ideals involved decrease progressively until they reach the lower bound, given by the smoothness bound used in the class group computations. We make use of the same result as in [Gél18] for the lattice reductions:

Theorem 6.1 ([Gél18, Theorem 4.3])**.** *The smallest vector v output by the BKZ algorithm with block-size β has a norm bounded by*

$$\|v\| \quad \leq \quad \beta^{\frac{n-1}{2(\beta-1)}} \cdot (\det \mathcal{L})^{\frac{1}{n}}.$$

The algorithm runs in time $\mathrm{Poly}(n, \log \|B_0\|) \left(\frac{3}{2}\right)^{\beta/2+o(\beta)}$*, with* B_0 *the input basis.*

We now fix the parameters for a degree-n number field $\mathbf{K}$ that belongs to a class $\mathcal{D}_{n_0,d_0,\alpha,\gamma}$ with $\frac{\gamma}{2} \leq \alpha \leq 2\gamma$. We know that the final complexity is given by $L_{|\Delta_\mathbf{K}|}\left(\frac{\alpha+\gamma}{3}\right)$, assuming this first constant is small enough — say below $\frac{1}{2}$. Let us write $k = \frac{\alpha+\gamma}{3}$ for the sake of simplicity. A pattern of the descent is displayed in Figure 2.

The initial reduction. Let $\mathfrak{a}$ be the ideal, assumed principal, for which we search for a generator. We may also assume that it is prime, otherwise it suffices to factor it and to work with the prime ideals, which have smaller norms. We can always represent this ideal with its HNF. We obtain an $n \times n$ matrix whose largest coefficient is at most the norm of the ideal $\mathcal{N}(\mathfrak{a})$.

The first reduction consists in performing a BKZ-reduction on the n-dimensional lattice $\varsigma(\mathfrak{a})$ with block-size $\beta = (\log |\Delta_\mathbf{K}|)^k$. It permits to exhibit a small vector v that satisfies $\|v\| \leq \beta^{\frac{n-1}{2(\beta-1)}} \mathcal{N}(\mathfrak{a})^{\frac{1}{n}}$, as $\det \varsigma(\mathfrak{a}) = \mathcal{N}(\mathfrak{a})$ (see Theorem 6.1). The cost of this lattice reduction is $L_{|\Delta_\mathbf{K}|}\big(k, o(1)\big)$, provided that the norm $\mathcal{N}(\mathfrak{a})$ satisfies $\log \mathcal{N}(\mathfrak{a}) \leq L_{|\Delta_\mathbf{K}|}(k - \varepsilon)$ for $\varepsilon > 0$. Therefore, the principal ideal generated by the algebraic integer $x_0 \in \mathfrak{a}$ corresponding to the vector $v \in \varsigma(\mathfrak{a})$ has its norm bounded by $(n+1)^n \cdot H(T)^n \cdot \beta^{\frac{n(n-1)}{2(\beta-1)}} \mathcal{N}(\mathfrak{a})$ (using the same technique as for Equation (3)). Finally, denoting by $\mathfrak{a}^{(0)}$ the unique integral ideal such that $\langle x_0 \rangle = \mathfrak{a} \cdot \mathfrak{a}^{(0)}$, we obtain the following upper bound:

$$\mathcal{N}\left(\mathfrak{a}^{(0)}\right) \leq L_{|\Delta_\mathbf{K}|}(\alpha + \gamma, n_0 d_0) = L_{|\Delta_\mathbf{K}|}(3k, n_0 d_0).$$

As we have mentioned, we alternate lattice reductions and smoothness tests. For keeping a complexity in $L_{|\Delta_\mathbf{K}|}(k)$, we are going to test the ideal $\mathfrak{a}^{(0)}$ for $L_{|\Delta_\mathbf{K}|}(2k, s_0)$-smoothness, for $s_0 > 0$ to be determined. Using ECM algorithm (see [Gél18, Appendix A]), the cost for a single test is $L_{|\Delta_\mathbf{K}|}\left(k, \sqrt{2ks_0}\right)$, while the assumption of Heuristic 3.3 asserts that the probability for $\mathfrak{a}^{(0)}$ to be $L_{|\Delta_\mathbf{K}|}(2k, s_0)$-smooth is lower bounded by $L_{|\Delta_\mathbf{K}|}\left(k, \frac{kn_0d_0}{s_0}\right)^{-1}$. First, this implies that we need to test on average $L_{|\Delta_\mathbf{K}|}(k)$ ideals before finding one that is smooth. We then make use of the randomization process used by Biasse and Fieker in [BF14]. It consists in considering randomized ideals that are products of $\mathfrak{a}$ with random power-products of small prime ideals — the ones in the factor base. Clearly, it offers sufficiently many choices for testing $L_{|\Delta_\mathbf{K}|}(k)$ ideals. Second, the total runtime for the smoothness tests is given by

$$L_{|\Delta_\mathbf{K}|}\left(k, \frac{kn_0d_0}{s_0} + \sqrt{2ks_0}\right),$$

which is minimal for $s_0^3 = 2k(n_0d_0)^2$, leading to a complexity of

$$L_{|\Delta_\mathbf{K}|}\left(k, \left(\frac{9}{2}k^2 n_0 d_0\right)^{\frac{1}{3}}\right).$$

Subsequent steps. At the beginning of the i-th step, we have an ideal $\mathfrak{a}^{(i)}$ whose norm is upper bounded by $L_{|\Delta_\mathbf{K}|}\left(k\left(1 + \frac{1}{2^i}\right), s_i\right)$. This time, we are going to perform the lattice reduction over a sublattice of $\varsigma\big(\mathfrak{a}^{(i)}\big)$ of dimension $d = c_d \left(\frac{\log |\Delta_\mathbf{K}|}{\log \log |\Delta_\mathbf{K}|}\right)^\delta$, for $0 \leq \delta \leq \alpha$ and $c_d > 0$ to be determined. The reason to look

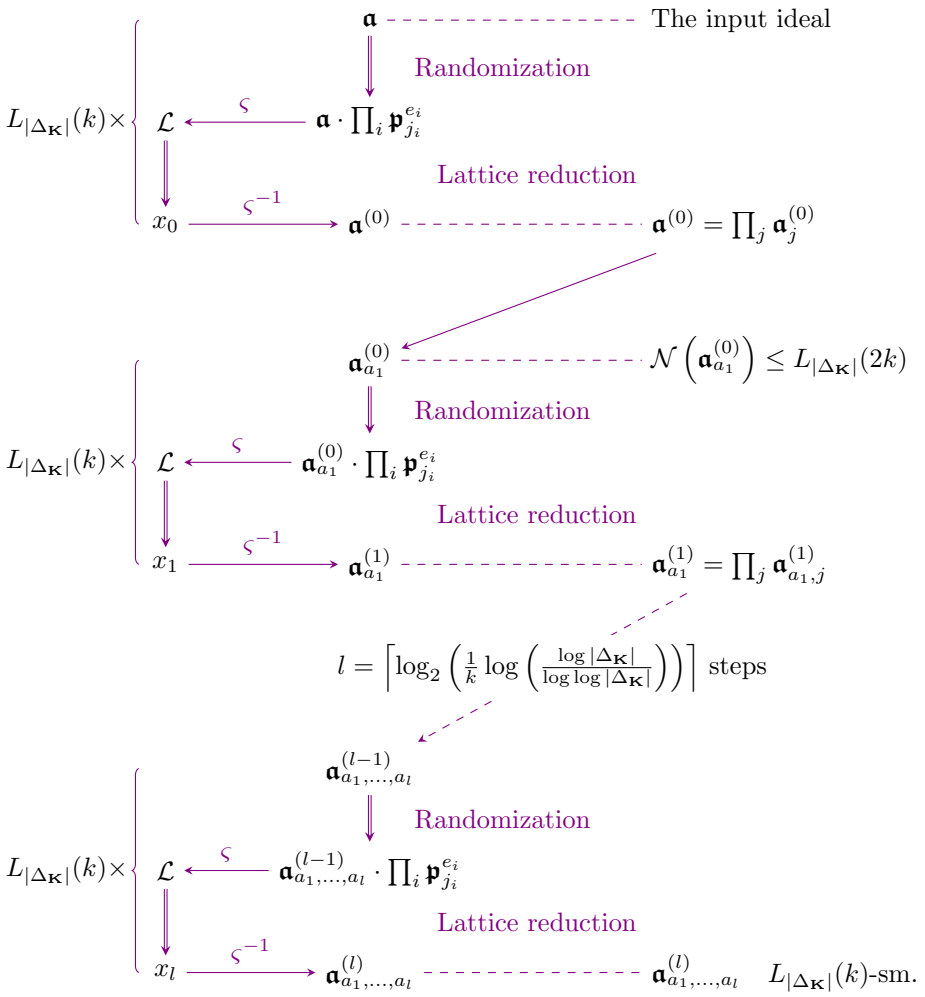

FIGURE 2. The descent algorithm for the medium-degree case.

at a sublattice is that it allows to reduce the norms of the ideals that are involved, which is exactly what we want for the descent.

The BKZ-reduction on this sublattice provides an algebraic integer $x_i \in \mathfrak{a}^{(i)}$ and so an integral ideal $\mathfrak{a}^{(i+1)}$ such that $\langle x_i\rangle = \mathfrak{a}^{(i)}\cdot\mathfrak{a}^{(i+1)}$. The upper bound we get on the norm of $\mathfrak{a}^{(i+1)}$, according to Theorem 6.1, is

$$L_{|\Delta_\mathbf{K}|}(\alpha)\cdot L_{|\Delta_\mathbf{K}|}\left(\gamma+\delta, d_0c_d\right)\cdot L_{|\Delta_\mathbf{K}|}(\alpha+\delta-k)\cdot L_{|\Delta_\mathbf{K}|}\left(\alpha+k\left(1+\frac{1}{2^i}\right)-\delta, \frac{n_0s_i}{c_d}\right).$$

This quantity is minimal when $\gamma+\delta=\alpha+k\left(1+\frac{1}{2^i}\right)-\delta \iff \delta=\alpha-k\left(1+\frac{1}{2^{i+1}}\right)$ and $c_d^2=\frac{n_0s_i}{d_0}$, which results in the following upper bound for the norm:

$$L_{|\Delta_\mathbf{K}|}\left(k\left(2+\frac{1}{2^{i+1}}\right), 2\sqrt{n_0d_0s_i}\right).$$

Again, we want to test this ideal for smoothness and we fix the smoothness bound to $L_{|\Delta_{\mathbf{K}}|}\left(k\left(1+\frac{1}{2^{i+1}}\right), s_{i+1}\right)$. This time, the cost for a single ECM is negligible, as given by $L_{|\Delta_{\mathbf{K}}|}\left(\frac{k}{2}\left(1+\frac{1}{2^{i+1}}\right)\right)$. The total cost is then inferred from the number of ideals we have to test. Using the same process as for the initial reduction and assuming Heuristic 3.3, this number is

$$L_{|\Delta_{\mathbf{K}}|}\left(k, \frac{2k\sqrt{n_0 d_0 s_i}}{s_{i+1}}\right).$$

The final step. We fix $l = \left\lceil \log_2\left(\frac{1}{k}\log\left(\frac{\log|\Delta_{\mathbf{K}}|}{\log\log|\Delta_{\mathbf{K}}|}\right)\right)\right\rceil$. Thus, at step l, we have ideals that are $L_{|\Delta_{\mathbf{K}}|}\left(k\left(1+\frac{1}{2^l}\right), s_l\right)$-smooth. However, by definition of the L-notation,

$$\log L_{|\Delta_{\mathbf{K}}|}\left(k\left(1+\frac{1}{2^l}\right), s_l\right) \leq s_l(\log|\Delta_{\mathbf{K}}|)^k(\log\log|\Delta_{\mathbf{K}}|)^{1-k} \times \underbrace{\left(\frac{\log|\Delta_{\mathbf{K}}|}{\log\log|\Delta_{\mathbf{K}}|}\right)^{1/\log\left(\frac{\log|\Delta_{\mathbf{K}}|}{\log\log|\Delta_{\mathbf{K}}|}\right)}}_{=\quad e=\exp(1)}\big(1+o(1)\big),$$

so that we have the inequality $L_{|\Delta_{\mathbf{K}}|}\left(k\left(1+\frac{1}{2^l}\right), s_l\right) \leq L_{|\Delta_{\mathbf{K}}|}(k, e\cdot s_l)$.

Remark 6.2. More precisely, we can go further and get rid of the constant e. Indeed, for every $\varepsilon > 0$, if C_ε denotes the smallest integer larger than $\log(1+\varepsilon)^{-1}$, then at step $C_\varepsilon \cdot l$, we only consider ideals that are $L_{|\Delta_{\mathbf{K}}|}(k, (1+\varepsilon)s_l)$-smooth.

In the end, we want all the ideals involved to have a norm below the smoothness bound we have used for class group computation, *i.e.,*

$$e \cdot s_l \leq c_b = \left(\frac{4k^2 n_0 d_0(\omega+1)}{\omega^2}\right)^{\frac{1}{3}}. \tag{12}$$

Our approach is to balance the cost of all steps, except the initial one: each one costs $L_{|\Delta_{\mathbf{K}}|}\left(k, \left(4k^2 n_0 d_0 y\right)^{\frac{1}{3}}\right)$, for a constant $y > 0$ to be determined. Hence we have, for all i,

$$\frac{2k\sqrt{n_0 d_0 s_i}}{s_{i+1}} = 4k^2 n_0 d_0 y \quad \Longleftrightarrow \quad s_{i+1} = \sqrt{s_i}\cdot\left(\frac{4k^2 n_0 d_0}{y^2}\right)^{\frac{1}{6}}.$$

We deduce that

$$\begin{aligned} s_l &= s_0^{\frac{1}{2^l}}\cdot\left(\frac{4k^2 n_0 d_0}{y^2}\right)^{\frac{1}{6}\cdot\left(1+\frac{1}{2}+\cdots+\frac{1}{2^{l-1}}\right)} \\ &= \left(\frac{s_0 y^{\frac{2}{3}}}{(4k^2 n_0 d_0)^{\frac{1}{3}}}\right)^{\frac{1}{2^l}}\left(\frac{4k^2 n_0 d_0}{y^2}\right)^{\frac{1}{3}} \\ &= \left(\frac{4k^2 n_0 d_0}{y^2}\right)^{\frac{1}{3}}\big(1+o(1)\big). \end{aligned}$$

Then, Equation (12) can be rewritten as $e\left(\frac{4k^2 n_0 d_0}{y^2}\right)^{\frac{1}{3}} \leq \left(\frac{4k^2 n_0 d_0(\omega+1)}{\omega^2}\right)^{\frac{1}{3}}$, *i.e.,* $y^2 \geq \frac{e^3\omega^2}{\omega+1}$. As the number of steps is polynomial in $\log|\Delta_{\mathbf{K}}|$, the total cost of

the l steps of the descent is $L_{|\Delta_{\mathbf{K}}|}\left(k, \left(4k^2 n_0 d_0 y\right)^{\frac{1}{3}}\right)$, with $y^2 = \frac{e^3\omega^2}{\omega+1}$. It outweighs the initial reduction, because $4y > \frac{9}{2}$ for $\omega \geq 2$.

Remark 6.3. We need to bound the numbers of ideals involved in order to be sure of our final complexity. At each step, we spend time $L_{|\Delta_{\mathbf{K}}|}(k)$ for the smoothness tests. It follows that the number of ideals in the decomposition is bounded by $O\left(\left(\frac{\log|\Delta_{\mathbf{K}}|}{\log\log|\Delta_{\mathbf{K}}|}\right)^k\right)$. During the descent, the number of ideals is then multiplied by this factor at each step. Finally, the number of ideals at step l is quasi-polynomial $O\left(\left(\frac{\log|\Delta_{\mathbf{K}}|}{\log\log|\Delta_{\mathbf{K}}|}\right)^k\right)^l$. In Figure 2, indices have been added to the ideals to illustrate this.

At this point, the only remaining part consists in finding out how to decompose these ideals over the principal ideals collected for building the relation matrix. This is done by solving a linear system $MX = Y$, where M is the relation matrix and Y the valuations vector of the smooth ideal. To be sure that this system has a solution, we need to have a relation matrix of *almost-full* rank. By this unusual term, we only mean that we want all ideals in the factor base involved in the relations, except the ones whose degree is larger than the bound c_t. Indeed, they do not appear in a relation because of the parameters we use, but we do not care as they do not arise either in the descent process — this is a consequence of the dimensions of the sublattices that we use. The runtime of this part is $L_{|\Delta_{\mathbf{K}}|}\left(k, 2c_b\right)$ as the matrix of relations is already in HNF.

Finally, we also have $y < \frac{(\omega+1)^4}{\omega^2}$, which means that the complexity for solving the Principal Ideal Problem is the same as the complexity obtained for class group computation. However, we have analyzed the runtime of the descent for the case when the matrix of relations is known.

Remark 6.4. Two improvements can be made to reduce the complexity. First, as explained in Remark 6.2, the constant e can be replaced by any other constant larger than and arbitrarily close to 1. Second, if we are only interested in solving the PIP, then the computation of the regulator and the class group structure are useless. Hence, the linear-algebra step boils down to solving a linear system over $\mathbf{Z}$, which can be performed in time $L_{|\Delta_{\mathbf{K}}|}\left(k, \omega c_b\right)$ using a Las-Vegas algorithm described by Storjohann in [Sto05]. Then, we can adjust all our parameters replacing $\omega+1$ by ω. Finally, these enhancements lead to a final complexity for the PIP of

$$L_{|\Delta_{\mathbf{K}}|}\left(k, \left(\frac{4k^2 n_0 d_0 \omega^4}{(\omega-1)^2}\right)^{\frac{1}{3}}\right).$$

Remark 6.5. The descent strategy for solving the Principal Ideal Problem is also treated in detail in [BEF+17]. It is applied in the context of the cryptanalysis of a Fully Homomorphic Encryption Scheme over prime-power cyclotomic fields. The interested reader can find more details there.

6.2. **The large-degree case.** For the present large-degree case, the approach is similar to the previous case, the only difference being the parameters choice. This time, $\alpha > 2\gamma$ and we denote by k the first constant of the class group complexity, *i.e.*, $k = \frac{\alpha}{2}$.

We perform the first reduction using a block-size $\beta = c_\beta(\log|\Delta_{\mathbf{K}}|)^k$. It still costs $L_{|\Delta_{\mathbf{K}}|}\big(k, o(1)\big)$ and gives rise to an algebraic integer x_0 and an integral ideal $\mathfrak{a}^{(0)}$ such that $\langle x_0\rangle = \mathfrak{a}\cdot\mathfrak{a}^{(0)}$. The norm of $\mathfrak{a}^{(0)}$ satisfies

$$\mathcal{N}\left(\mathfrak{a}^{(0)}\right) \leq L_{|\Delta_{\mathbf{K}}|}\left(2\alpha - k, \frac{n_0^2}{2c_\beta}\right) = L_{|\Delta_{\mathbf{K}}|}\left(3k, \frac{n_0^2}{2c_\beta}\right).$$

We make use of the same randomization process as in the medium-case and obtain a $L_{|\Delta_{\mathbf{K}}|}(2k, s_0)$-smooth ideal in time $L_{|\Delta_{\mathbf{K}}|}\left(k, \left(\frac{9k^2n_0^2}{4c_\beta}\right)^{\frac{1}{3}}\right)$, for $s_0^3 = \frac{kn_0^4}{2c_\beta^2}$ chosen to minimize this cost.

The subsequent steps begin with an ideal of norm less than $L_{|\Delta_{\mathbf{K}}|}\left(k\left(1+\frac{1}{2^i}\right), s_i\right)$. Then, by fixing $\delta = k\left(1+\frac{1}{2^{i+1}}\right)$, we obtain an ideal $\mathfrak{a}^{(i+1)}$ such that is norm is upper-bounded by

$$L_{|\Delta_{\mathbf{K}}|}\left(k\left(2+\frac{1}{2^{i+1}}\right), \frac{n_0 s_i}{c_d}\right).$$

so that, assuming Heuristic 3.3, we can find a $L_{|\Delta_{\mathbf{K}}|}\left(k\left(1+\frac{1}{2^{i+1}}\right), s_{i+1}\right)$-smooth ideal in time

$$L_{|\Delta_{\mathbf{K}}|}\left(k, \frac{kn_0 s_i}{c_d s_{i+1}}\right).$$

In the same way, setting $l = \left\lceil \log_2\left(\frac{1}{k}\log\left(\frac{\log|\Delta_{\mathbf{K}}|}{\log\log|\Delta_{\mathbf{K}}|}\right)\right)\right\rceil$ implies that after step l, the ideals involved are $L_{|\Delta_{\mathbf{K}}|}(k, e\cdot s_l)$-smooth; here we want $e\cdot s_l$ to be smaller than c_b.

Let $y > 0$ be a constant such that, at each step, the runtime of the smoothness tests is below $L_{|\Delta_{\mathbf{K}}|}(k, y)$. That means that for all i, it is the case that $\frac{kn_0 s_i}{c_d s_{i+1}} \leq y$. Then, by fixing $c_d = \frac{kn_0}{y}\cdot\left(\frac{es_0}{c_b}\right)^{\frac{1}{l}}$ and $s_{i+1} = s_i\cdot\left(\frac{c_b}{es_0}\right)^{\frac{1}{l}}$, the previous equation is satisfied, resulting in

$$s_l = s_0\cdot\left(\frac{c_b}{es_0}\right) \qquad \Longleftrightarrow \qquad e\cdot s_l = c_b.$$

As $y > 0$ can be chosen arbitrarily small, each step has a runtime in $L_{|\Delta_{\mathbf{K}}|}\big(k, o(1)\big)$ and the initial-reduction cost can also be chosen that small, for c_β sufficiently large. The remaining part consisting in solving the linear system works in the same way as for the previous case and we can conclude that the complexity of our algorithm for solving the PIP is the same as the complexity of the class group computation. Again, Remark 6.4 holds so that we can reduce the complexity to

$$L_{|\Delta_{\mathbf{K}}|}\left(k, \left(\frac{k^2 n_0\omega^2}{2(\omega-1)}\right)^{\frac{1}{2}}\right).$$

6.3. **The small-degree case.** Again, we only give a brief summary of the descent. Here we have $2\alpha < \gamma$ and k denotes $\frac{\gamma}{2}$.

The initial BKZ-reduction provides an ideal of norm below $L_{|\Delta_{\mathbf{K}}|}(\alpha+\gamma, n_0 d_0)$ in time $L_{|\Delta_{\mathbf{K}}|}\big(k, o(1)\big)$. We can find an ideal that is $L_{|\Delta_{\mathbf{K}}|}(k+\alpha, s_0)$-smooth in time $L_{|\Delta_{\mathbf{K}}|}\left(k, \frac{kn_0 d_0}{s_0}\right)$ as the cost of a single application of ECM is negligible — because $\frac{k+\alpha}{2} < k$.

Then, every subsequent step takes as input an ideal of norm upper bounded by $L_{|\Delta_\mathbf{K}|}\left(k+\frac{\alpha}{2^i}, s_i\right)$. Then, looking for a small vector in the sublattice of dimension $d = c_d\left(\frac{\log|\Delta_\mathbf{K}|}{\log\log|\Delta_\mathbf{K}|}\right)^{\frac{\alpha}{2^{i+1}}}$ leads to a new ideal whose norm is smaller than $L_{|\Delta_\mathbf{K}|}\left(2k+\frac{\alpha}{2^{i+1}}, d_0 c_d\right)$. Again we expect, assuming Heuristic 3.3, to find one that is $L_{|\Delta_\mathbf{K}|}\left(k+\frac{\alpha}{2^{i+1}}, s_{i+1}\right)$-smooth in time $L_{|\Delta_\mathbf{K}|}\left(k, \frac{kd_0c_d}{s_{i+1}}\right)$.

At final step $l = \left\lceil \log_2\left(\frac{1}{\alpha}\log\left(\frac{\log|\Delta_\mathbf{K}|}{\log\log|\Delta_\mathbf{K}|}\right)\right)\right\rceil$, we have $L_{|\Delta_\mathbf{K}|}(k, e\cdot s_l)$-smooth ideals and we want $e\cdot s_l$ to be smaller than $c_b = \left(\frac{kd_0c_t}{\omega}\right)^{\frac{1}{2}}$. Note that, at this point, $d = c_d e\big(1+o(1)\big)$ for the same reason as above. Hence c_d may be as small as $\frac{1}{e}$ and the cost of the final smoothness test is lower-bounded by

$$L_{|\Delta_\mathbf{K}|}\left(k, \frac{kd_0}{e\cdot s_l}\right) \geq L_{|\Delta_\mathbf{K}|}\left(k, \frac{kd_0}{c_b}\right) = L_{|\Delta_\mathbf{K}|}\left(k, \left(\frac{kd_0\omega}{c_t}\right)^{\frac{1}{2}}\right).$$

This last smoothness test dominates the overall complexity of the descent phase, as we can always choose c_d and x_i such that the runtimes of the other smoothness tests become arbitrarily small. In addition, this part is dominated by the class group computation: indeed, $\left(\frac{kd_0\omega}{c_t}\right)^{\frac{1}{2}} \leq (\omega+1)\left(\frac{kd_0c_t}{\omega}\right)^{\frac{1}{2}}$ because $c_t \geq 1 > \frac{w}{w+1}$. Again, we can improve this algorithm as explained in Remark 6.4 and finally get a complexity of

$$L_{|\Delta_\mathbf{K}|}\left(k, \left(\frac{kd_0c_t\omega^2}{\omega-1}\right)^{\frac{1}{2}}\right).$$

Remark 6.6. Thanks to this precise analysis, we are able to derive a precise complexity estimate of the attack presented in [BEF+17]. Indeed, prime-power cyclotomic fields — together with their totally real subfields — asymptotically belong to the class $\mathcal{D}_{1,0,1,1}$. Then the result stated at the very end of Section 6.2 implies that the complexity of this attack can be as low as

$$L_{|\Delta_\mathbf{K}|}\left(\frac{1}{2}, \frac{\omega}{2\sqrt{2(\omega-1)}}\right).$$

Taking $\omega = \log_2 7$, we obtain a runtime for our attack of

$$L_{|\Delta_\mathbf{K}|}\left(\frac{1}{2}, 0.738\right) = 2^{1.066\cdot n^{\frac{1}{2}}\log n}.$$

References

[BEF+17] Jean-François Biasse, Thomas Espitau, Pierre-Alain Fouque, Alexandre Gélin, and Paul Kirchner, *Computing generator in cyclotomic integer rings - A subfield algorithm for the Principal Ideal Problem in L(1/2) and application to the cryptanalysis of a FHE scheme*, Advances in Cryptology - EUROCRYPT 2017, Proceedings, 2017, pp. 60–88.

[BF14] Jean-François Biasse and Claus Fieker, *Subexponential class group and unit group computation in large degree number fields*, LMS Journal of Computation and Mathematics **17** (2014), 385–403.

[Bia14] Jean-François Biasse, *An L(1/3) algorithm for ideal class group and regulator computation in certain number fields*, Mathematics of Computation **83** (2014), 2005–2031.

[BJN+99] Johannes Buchmann, Michael J. Jacobson, Stefan Neis, Patrick Theobald, and Damian Weber, *Sieving methods for class group computation*, Algorithmic Algebra and Number Theory, Proceedings, 1999, pp. 3–10.

[BL10] Yuval Bistritz and Alexander Lifshitz, *Bounds for resultants of univariate and bivariate polynomials*, Linear Algebra and its Applications **432** (2010), 1995–2005.

[Buc90] Johannes Buchmann, *A subexponential algorithm for the determination of class groups and regulators of algebraic number fields*, Séminaire de Théorie des Nombres, Paris 1988-1989 (1990), 27–41.

[CDPR16] Ronald Cramer, Léo Ducas, Chris Peikert, and Oded Regev, *Recovering short generators of principal ideals in cyclotomic rings*, Advances in Cryptology - EUROCRYPT 2016, Proceedings, 2016, pp. 559–585.

[Coh93] Henri Cohen, *A course in computational algebraic number theory*, Graduate Texts in Mathematics, vol. 138, Springer-Verlag, New-York, 1993.

[EG07] Andreas Enge and Pierrick Gaudry, *An L(1/3+ε) algorithm for the discrete logarithm problem for low degree curves*, Advances in Cryptology - EUROCRYPT 2007, Proceedings, 2007, pp. 379–393.

[EGT11] Andreas Enge, Pierrick Gaudry, and Emmanuel Thom, *An L(1/3) discrete logarithm algorithm for low degree curves*, Journal of Cryptology **24** (2011), 24–41.

[Gél18] Alexandre Gélin, *On the complexity of class group computations for large-degree number fields*, arXiv:1810.11396, 2018, `https://arxiv.org/pdf/1810.11396.pdf`.

[Gen09] Craig Gentry, *Fully homomorphic encryption using ideal lattices*, Proceedings of the 41st Annual ACM Symposium on Theory of Computing STOC 2009, 2009, pp. 169–178.

[GGH13] Sanjam Garg, Craig Gentry, and Shai Halevi, *Candidate multilinear maps from ideal lattices*, Advances in Cryptology - EUROCRYPT 2013, Proceedings, 2013, pp. 1–17.

[GJ16] Alexandre Gélin and Antoine Joux, *Reducing number field defining polynomials: an application to class group computation*, LMS Journal of Computation and Mathematics **19** (2016), 315–331.

[HM89] James L. Hafner and Kevin S. McCurley, *A rigorous subexponential algorithm for computation of class groups*, Journal of American Mathematical Society **2** (1989), 839–850.

[Lan03] Edmund Landau, *Neuer Beweis des Primzahlsatzes und Beweis des Primidealsatzes*, Mathematische Annalen **56** (1903), 645–670.

[LLMP90] Arjen K. Lenstra, Hendrik W. Lenstra Jr., Mark S. Manasse, and John M. Pollard, *The number field sieve*, Proceedings of the 22nd Annual ACM Symposium on Theory of Computing STOC 1990, 1990, pp. 564–572.

[Sha69] Daniel Shanks, *Class number, a theory of factorization, and genera*, Proceedings of Symposia in Pure Mathematics, vol. 20, 1969, pp. 415–440.

[Sha72] ———, *The infrastructure of a real quadratic field and its applications*, Proceedings of the 1972 Number Theory Conference, 1972, pp. 217–224.

[Sto05] Arne Storjohann, *The shifted number system for fast linear algebra on integer matrices*, Journal of Complexity **21** (2005), no. 4, 609–650.

[SV10] Nigel P. Smart and Frederik Vercauteren, *Fully homomorphic encryption with relatively small key and ciphertext sizes*, Public Key Cryptography - PKC 2010, Proceedings, 2010, pp. 420–443.

[Thi95] Christoph Thiel, *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`.

Laboratoire de Mathématiques de Versailles, UVSQ, CNRS, Université Paris-Saclay, Versailles, France

E-mail address: `alexandre.gelin@uvsq.fr`