LMS J. Comput. Math. 17 (Special issue A) (2014) 385–403
doi:10.1112/S1461157014000345

Subexponential class group and unit group computation in large degree number fields

Jean-François Biasse and Claus Fieker

Abstract

We describe how to compute the ideal class group and the unit group of an order in a number field in subexponential time. Our method relies on the generalized Riemann hypothesis and other usual heuristics concerning the smoothness of ideals. It applies to arbitrary classes of number fields, including those for which the degree goes to infinity.

1. *Introduction*

Let $K = \mathbb{Q}(\theta)$ be a number field of degree n and maximal order $\mathcal{O}_K$. The ideal class group of an order $\mathcal{O}$ of K of discriminant $\Delta = \mathrm{disc}(\mathcal{O})$ is a finite Abelian group that can be decomposed as $\mathrm{Cl}(\mathcal{O}) = \bigoplus_i \mathbb{Z}/d_i\mathbb{Z}$, with $d_i \mid d_{i+1}$. Computing the structure of $\mathrm{Cl}(\mathcal{O})$, together with a system of fundamental units of $\mathcal{O}$, is a major task in computational number theory.

In 1968, Shanks [**31**, **32**] proposed an algorithm relying on the baby-step giant-step method to compute the class number and the regulator of a quadratic number field in time $O(|\Delta|^{1/4+\epsilon})$, or $O(|\Delta|^{1/5+\epsilon})$ under the extended Riemann hypothesis [**24**]. Then, a subexponential strategy for the computation of the group structure of the class group of an imaginary quadratic extension was described in 1989 by Hafner and McCurley [**17**]. The expected running time of this method is

$$L_\Delta(\tfrac{1}{2}, \sqrt{2} + o(1)) = e^{(\sqrt{2}+o(1))\sqrt{\log|\Delta| \log\log|\Delta|}}.$$

Buchmann [**9**] generalized this result to the case of an arbitrary extension, thus obtaining a heuristic complexity bounded by $L_\Delta(\frac{1}{2}, 1.7 + o(1))$. This complexity is valid for fixed degree n and Δ tending to infinity. Practical improvements to Buchmann's algorithm were presented in [**12**] by Cohen, Diaz Y Diaz and Olivier. More recently, Biasse described an algorithm for computing the ideal class group and the unit group of $\mathcal{O} = \mathbb{Z}[\theta]$ in heuristic complexity bounded by $L_\Delta(\frac{1}{3}, c)$ for some $c > 0$ valid in certain classes of number fields.

The computation of the class group and unit group of an order in classes of large degree recently received a growing attention due to their connection with cryptosystems based on the hardness of finding short vectors in ideal lattices, such as the homomorphic encryption scheme of Vercauteren and Smart [**33**]. Also, ideal decomposition in orders of complex multiplication (CM) fields allows us to compute isogenies between Abelian varieties by the method described in the elliptic case by Jao and Soukharev [**20**].

Contribution. Our main result is the first heuristic subexponential algorithm for ideal class group and unit group computation in an order of an arbitrary class of number fields, including when the degree is large, as opposed to all previous algorithms. We achieve an $L_\Delta(\frac{2}{3} + \varepsilon, c)$ complexity for arbitrarily small ε and $c > 0$ for all classes of number fields. Under certain restrictions, we obtain an $L_\Delta(a, c)$ algorithm for $\frac{1}{3} \leqslant a < \frac{2}{3}$. We carefully address the numerical

Received 27 February 2014; revised 23 May 2014.

2010 Mathematics Subject Classification 54C40, 14E20 (primary), 46E25, 20C20 (secondary).

Contributed to the Algorithmic Number Theory Symposium XI, GyeongJu, Korea, 6–11 August 2014.

precision issues by using an original p-adic method inspired by [7], and we show how to return a compact representation of the units in subexponential time as well, while all the previous methods were exponential in the degree [10, 35] of the field. The present contribution is essentially theoretical, and focuses on the complexity analysis. The p-adic method and the compact representation algorithm were inspired by [7] while the relation search significantly differs from [7]. The results of [7] and this paper are not comparable because [7] consists of practical improvements on subexponential methods for fixed degree classes of number fields [9, 12] while here we consider classes of number fields of degree going to infinity.

The main ingredient that allows us to achieve a subexponential complexity even when the degree of the number field is large is the use of a different lattice reduction method when reducing ideals. More specifically, the LLL-reduction used to find a reduced ideal in [12] does not allow relations between ideals to be found in subexponential time, while the BKZ_k reduction [28] with a suitable k does. The use of p-adic logarithms to compute the unit group greatly simplifies the complexity analysis and the estimation of the precision loss during arithmetic operations. In addition, it allows the parallelization of this process. Finally, our compact representation method runs in subexponential time even when the degree in large because we rely on LLL-reductions while the other methods [10, 35] need exact solutions to instances of the shortest vector problem, which is exponential in the degree.

2. *General description of the index calculus method*

The subexponential method due to Buchmann [9] is a generalization of the algorithm of Hafner and McCurley [17] for quadratic number fields, and its complexity is subexponential bounded by $L_\Delta(\frac{1}{2}, O(1))$, for classes of number fields with fixed degree. It relies on an index calculus strategy. Let $\mathcal{B} = \{\mathfrak{p}_1, \dots, \mathfrak{p}_N\}$ be a set of invertible prime ideals of an order $\mathcal{O}$ whose classes generate $\mathrm{Cl}(\mathcal{O})$. We have a surjective morphism

$$\begin{array}{ccccc} \mathbb{Z}^N & \xrightarrow{\varphi} & \mathcal{I} & \xrightarrow{\pi} & \mathrm{Cl}(\mathcal{O}) \\ (e_1, \dots, e_N) & \longrightarrow & \prod_i \mathfrak{p}_i^{e_i} & \longrightarrow & \prod_i [\mathfrak{p}_i]^{e_i}, \end{array}$$

and the class group $\mathrm{Cl}(\mathcal{O})$ is isomorphic to $\mathbb{Z}^N/\ker(\pi \circ \varphi)$. Therefore, computing the class group boils down to computing $\ker(\pi \circ \varphi)$, which is given by the lattice of $(e_1, \dots, e_N) \in \mathbb{Z}^N$ such that $\mathfrak{p}_1^{e_1}, \dots, \mathfrak{p}_N^{e_N} = (\alpha)$, where $\alpha \in \mathcal{O}$. We collect many relations of the form $\prod_i \mathfrak{p}_i^{e_i^{(j)}} = (\alpha_j)$ and put them in the rows of the relation matrix $M := (e_i^{(j)})$. If we have found enough of them, they give us the group structure of $\mathrm{Cl}(\mathcal{O})$ via the Smith normal form (SNF) of M. Meanwhile, every vector $X := (x_1, \dots, x_N)$ of the left kernel of M yields a unit $\gamma_X := \alpha_1^{x_1} \dots \alpha_N^{x_N}$. Since the principal ideals that it generates satisfy

$$(\gamma_X) = \mathfrak{p}_1^{\sum_i x_i e_i^{(1)}} \dots \mathfrak{p}_N^{\sum_i x_i e_i^{(N)}} = \mathfrak{p}_1^0 \dots \mathfrak{p}_N^0 = (1) = \mathcal{O},$$

this allows us to iteratively compute the unit group U by finding kernel vectors of M.

Algorithm 1. Class group and unit group of $\mathcal{O}$.

Input: $\mathcal{O}$, K, $\mathcal{B} = \{\mathfrak{p} \subseteq \mathcal{O} \mid \mathcal{N}(\mathfrak{p}) \leqslant B\}$ that generates $\mathrm{Cl}(\mathcal{O})$.

Output: Class group and unit group of $\mathcal{O}$.

1: Derive random relations in $\mathrm{Cl}(\mathcal{O})$ between classes of elements of $\mathcal{B}$.
2: Let M be a matrix for a generating system of the $\mathbb{Z}$-lattice of the relations.
3: Compute $\ker(M)$.
4: Find a minimal generating set of the units of the form γ_X for $X \in \ker(M)$.
5: Find the group structure of $\mathrm{Cl}(\mathcal{O})$ from the SNF of M.
6: Certify the result.

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

3. *Finding relations when $n \to \infty$*

In this section, we present our method for deriving relations in $\mathrm{Cl}(\mathcal{O})$ where $\mathcal{O}$ is an order in a degree n number field K. Given a smoothness bound $B > 0$ and $\mathfrak{a} \subseteq \mathcal{O}$, we want to find products of the form $\mathfrak{a} = (\phi)\mathfrak{p}_1 \dots \mathfrak{p}_k$, where $\alpha \in \mathcal{O}$ and the $\mathfrak{p}_i$ are invertible prime ideals of $\mathcal{O}$ with $\mathcal{N}(\mathfrak{p}_i) \leqslant B$. We want our method to run in subexponential time $L_\Delta(a, c_1)$ for some $c_1 > 0$ and $0 < a < 1$ when B is a subexponential bound. Here Δ is the discriminant of $\mathcal{O}$ and the subexponential function is given by

$$L_\Delta(a, c) := e^{c \log(|\Delta|)^a \log\log(|\Delta|)^{1-a}}.$$

One way of finding relations between ideals in $\mathrm{Cl}(\mathcal{O})$ is to enumerate random B-smooth ideals in $\mathcal{O}$ (which means that they are power-products of prime ideals of norm bounded by B) until one is equivalent to another B-smooth ideal (this test usually involves reducing it first, as explained in § 3.3). The other typical method to find relations is to enumerate elements $\phi \in \mathcal{O}$ until the principal ideal it generates is B-smooth. In one case, it is the probability of the smoothness of an ideal which rules the run time of an algorithm, and in the other case, it is the probability of smoothness of an element, which is much less understood, in particular due to the units of $\mathcal{O}$.

3.1. *Smoothness of ideals*

In [**30**], Scourfield established a result on the smoothness of ideals in a number field comparable to the ones known on integers. Let

$$\Psi(x, y) := |\{\mathfrak{a} \subseteq \mathcal{O}_K \mid \mathcal{N}(\mathfrak{a}) \leqslant x, \mathfrak{a} \text{ } y\text{-smooth}\}|,$$

and $\varepsilon > 0$, then $\Psi(x, y)/x \sim \lambda_K \rho(u)$, where $u = \log(x)/\log(y)$, ρ is the Dickman function, λ_K is the residue of the zeta function $\zeta_K(s)$ at $s = 1$ and $(\log\log(x))^{(5/3)+\varepsilon} \leqslant \log(y) \leqslant \log(x)$, $x \geqslant x_0(\varepsilon)$ for some $x_0(\varepsilon)$. In the case where K is normal and $n/(\log|\mathrm{disc}\,\mathcal{O}_K|) \to 0$, λ_K can be bounded absolutely, but there is no such result in the general case. We therefore rely on the following heuristic for the smoothness of ideals.

Heuristic 1. We assume that the probability $P(\iota, \mu)$ that an ideal of $\mathcal{O}$ of norm bounded by ι is a power-product of prime ideals of norm bounded by μ satisfies

$$P(\iota, \mu) \geqslant e^{(-u \log u(1+o(1)))} \quad \text{for } u = \log(\iota)/\log(\mu). \tag{3.1}$$

We do not know if Scourfield's result remains true when we restrict ourselves to principal ideals. This is one of the reasons why the complexity of the number field sieve [**23**] (NFS) is only heuristic. In addition to the fact that we assume the same probability of smoothness for principal ideals as for general ideals, we require a stronger smoothness assumption to perform our $\mathfrak{q}$-descent that is described in § 3.3. Indeed, we want the primes in our decomposition to be of inertia degree 1, that is of the form $p\mathcal{O} + (\theta - v_p)\mathcal{O}$, where v_p is a root of $T(X)$ mod p. In general, prime ideals can have inertia degree $f \geqslant 2$ and thus be of the form $p\mathcal{O} + T_p(\theta)\mathcal{O}$ where $\deg(T_p) = f$. However, their proportion is low when $B = L_\Delta(a, c)$ for some $0 < a < 1$ and $c > 0$. For $2 \leqslant f \leqslant n$, we have

$$\#\{p \text{ prime} \mid p^f \leqslant B\} \sim \frac{B^{1/f}}{\log(B^{1/f})} = \frac{fB^{1/f}}{\log B}.$$

The proportion of primes whose fth power for $2 \leqslant f \leqslant n$ is below the smoothness bound B with respect to the primes bounded by B thus equals

$$\frac{1}{\pi(B)} \sum_{2 \leqslant f \leqslant n} \frac{fB^{1/f}}{\log B} = \sum_{2 \leqslant f \leqslant n} \frac{1}{L_\Delta(a, c - c/f + o(1))} \leqslant \frac{1}{L_\Delta(a, c/2 + o(1))},$$

since n is polynomial in $\log|\Delta|$.

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

HEURISTIC 2. We assume that the probability $P(\iota, \mu)$ that a principal ideal of $\mathcal{O}$ of norm bounded by ι is a power-product of degree 1 prime ideals of norm bounded by μ satisfies

$$P(\iota, \mu) \geqslant e^{(-u \log u(1+o(1)))} \quad \text{for } u = \log(\iota)/\log(\mu). \tag{3.2}$$

COROLLARY 3.1. *Let* $\iota = \lfloor \log L_\Delta(\zeta, c) \rfloor$ *and* $\mu = \lceil \log L_\Delta(\beta, d) \rceil$. *Then assuming Heuristic* 1 *or Heuristic* 2, *we have*

$$P(\iota, \mu) \geqslant L_\Delta\left(\zeta - \beta, \frac{-c}{d}(\zeta - \beta) + o(1)\right).$$

3.2. *The BKZ-reduction*

Given an ideal $\mathfrak{a} \subseteq \mathcal{O}$ and $B > 0$, the classic method derived from [9, 17] to produce a relation of the form $\mathfrak{a} = (\phi)\mathfrak{p}_1^{e_1}, \ldots, \mathfrak{p}_N^{e_N}$ consists of choosing $\mathcal{B} = \{\mathfrak{p} \mid \mathcal{N}(\mathfrak{p}) \leqslant B\}$ where $B \leqslant L_\Delta(1/2, O(1))$ and testing random ideals of the form $\mathfrak{a} \cdot \prod_i \mathfrak{p}_i^{e_i}$ where $\mathcal{N}(\mathfrak{p}_i) \leqslant 48(\log(|\Delta|))^2$ and $e_i \leqslant |\Delta|$ for B-smoothness in $\mathrm{Cl}(\mathcal{O})$. Indeed, under the generalized Riemann hypothesis (GRH), the classes of ideals of norm less than $48(\log(|\Delta|))^2$ generate the class group whose size is bounded by $\sqrt{|\Delta|}$. This is a direct consequence of [2, Theorem 4], which states that the primes of norm up to $12\log^2(|\mathrm{disc}(\mathcal{O}_K)|^2\mathcal{N}(\mathfrak{f}))$ generate the ray class group of conductor $\mathfrak{f}$. There is an $\mathfrak{f}$ such that $\mathrm{Cl}(\mathcal{O})$ is a quotient of the ray class group of conductor $\mathfrak{f}$ and we have

$$12\log^2(|\mathrm{disc}(\mathcal{O}_K)|^2\mathcal{N}(\mathfrak{f})) \leqslant 12\log^2(|\mathrm{disc}(\mathcal{O}_K)|^2\mathcal{N}(\mathfrak{f})^4) = 12\log^2(|\Delta|^2) = 48(\log(|\Delta|))^2.$$

A reduction precedes the test for smoothness of a power-product of ideals $\mathfrak{a} \cdot \prod_i \mathfrak{p}_i^{e_i}$ to find an ideal $\mathfrak{b} \subseteq \mathcal{O}$ in the same equivalence class as $\mathfrak{a}' := \mathfrak{a} \cdot \prod_i \mathfrak{p}_i^{e_i}$ with a reasonably bounded norm. It is done by finding a short element $\phi \in \mathfrak{c}$ where $\mathfrak{a}'^{-1} = (1/l)\mathfrak{c}$ with $l \in \mathbb{Z}_{>0}$ and $\mathfrak{c} \subseteq \mathcal{O}$. Such a short element satisfies $\|\phi\| \leqslant \lambda_\mathcal{O}|\Delta|^{1/2n}\mathcal{N}(\mathfrak{c})^{1/n}$, where $\lambda_\mathcal{O}$ is an approximation factor depending on the reduction method that we use.

With the LLL algorithm, we have $\lambda_\mathcal{O} = 2^{n/2}$ achieved in polynomial time in n. As shown in [5], this does not allow a subexponential relation collection phase when $n \to \infty$. Instead, there are lattice reduction methods that offer the possibility of a trade-off between the time spent in the reduction and the approximation factor $\lambda_\mathcal{O}$. One can use Schnorr's algorithm [29], which was later improved by Gama and Nguyen [15], or the BKZ algorithm originally described by Schnorr and Euchner [28], and rigorously analyzed by Hanrot, Pujol and Stehlé [18], relying on the shortest vector subroutine described in [25]. Both are equivalent for our purposes, but BKZ is known to outperform Gama and Nguyen's method in practice. The BKZ algorithm with parameter k relies on finding the shortest vector in k-dimensional blocks of the original lattice. Using BKZ with parameter k, it is possible to find an approximate short vector with $\lambda_\mathcal{O} = k^{n/2k}$ in time $2^{O(k)} \times P(n)$ where P is a polynomial. We describe in Algorithm 2 how to use it to reduce ideals.

ALGORITHM 2. BKZ-reduction.

Input: An ideal $\mathfrak{a} \subseteq \mathcal{O}$, and $k \geqslant 1$.
Output: $\mathfrak{b} \subseteq \mathcal{O}$ and $\phi \in K$ with $\mathcal{N}(\mathfrak{b}) \leqslant k^{n^2/2k}\sqrt{|\Delta|}$, where $\mathfrak{b} = (\phi)\mathfrak{a}$, $n = \dim(\mathcal{O})$ and $\Delta = \mathrm{disc}(\mathcal{O})$.

1: $\mathfrak{c} \leftarrow l\mathfrak{a}^{-1}$ where l is the denominator of $\mathfrak{a}$.
2: Find a BKZ$_k$-reduced $\gamma \in \mathfrak{c}$.
3: $\mathfrak{b} \leftarrow (\gamma/l)\mathfrak{a}$.
4: **return** $\mathfrak{b}, \gamma/l$.

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

ALGORITHM 3. Ideal decomposition with the BKZ-reduction.

Input: Ideal $\mathfrak{a}$, $\varepsilon > 0$ and $B > 0$.
Output: Primes $\mathfrak{q}_i$ with $\mathcal{N}(\mathfrak{q}_i) \leqslant B$, $\phi \in K$ such that $\mathfrak{a} = (\phi) \cdot \prod_i \mathfrak{q}_i$.

$k \leftarrow \log(|\Delta|)^{\varepsilon}$.
$\mathfrak{a} \leftarrow (\phi_1)\mathfrak{a}$ where ϕ_1 is the output of Algorithm 2 on $(\mathfrak{a}, k)$.
found $\leftarrow$ false
while found = false **do**
  Let $\mathfrak{q}_1, \ldots, \mathfrak{q}_N$ be random prime ideals with $\mathcal{N}(\mathfrak{q}_i) \leqslant 48 \log(|\Delta|)^2$.
  $\mathfrak{a}' \leftarrow (\phi_i) \cdot \mathfrak{a} \prod_i \mathfrak{q}_i^{-1}$ where ϕ_2 is the output of Algorithm 2 on $(\mathfrak{a} \cdot \prod_i \mathfrak{q}_i^{-1}, k)$.
  if $\mathfrak{a}'$ is B-smooth **then**
    found $\leftarrow$ true
    Let β and $(\mathfrak{p}_j)$ such that $\mathfrak{a}' = (\beta) \prod_j \mathfrak{p}_j$.
  end if
end while

return $\{(\mathfrak{q}_i)_{i \leqslant N}, (\mathfrak{p}_j)\}$, $\phi_1 \cdot \phi_2 \cdot \beta$

PROPOSITION 3.2 (GRH + Heuristic 1). *Let $\mathcal{O}$ be an order in a degree n number field, $\Delta = \operatorname{disc}(\mathcal{O})$, $B > 0$, and $\mathfrak{a} \subseteq \mathcal{O}$ an ideal. Then Algorithm 3 returns a B-smooth decomposition in time*

$$\log(\mathcal{N}(\mathfrak{a}))^{1+o(1)} L_{\Delta}(a, c_1)$$

where:

$B = L_{\Delta}(a, c_2)$ *for some* $c_1, c_2 > 0$;
$a = \frac{2}{3} + \varepsilon$ *for* ε *arbitrarily small in the general case;*
$a = \frac{1}{2}$ *when* $n \leqslant \log(|\Delta|)^{3/4-\varepsilon}$ *for* ε *arbitrarily small.*

The proof of Proposition 3.2 can be found in [5].

3.3. *The $\mathfrak{q}$-descent*

The $\mathfrak{q}$-descent is a strategy that allows a complexity bounded by $L_{\Delta}(a, b)$ for $a < \frac{1}{2}$ and $b > 0$ in certain classes of orders. The generalization of Buchmann's method together with a BKZ-reduction can only yield an $L_{\Delta}(\frac{1}{2})$ algorithm for computing $\mathrm{Cl}(\mathcal{O})$ and solving related problems. Indeed, no matter how small the approximation factor $\lambda_{\mathcal{O}}$ is, the norm of the reduced ideal cannot have a tighter bound than $|\Delta|^{O(1)}$. The idea of the $\mathfrak{q}$-descent derives from the algorithms based on the number field sieve [23] to solve the discrete logarithm problem in time $L_q(\frac{1}{3})$ in $\mathbb{F}_q$ (see in particular [1, 16, 22]). Our method is directly inspired by the analogue for C_{ab} curves presented in [13].

A version of the $\mathfrak{q}$-descent was considered in [6] to derive relations between ideals of norm bounded by $L_{\Delta}(\frac{1}{3}, c_1)$ in time $L_{\Delta}(\frac{1}{3}, c_2)$ in the equation order $\mathbb{Z}[\theta]$ of a number field $\mathbb{Q}(\theta)$ with degree and height of defining polynomial growing in certain proportions. It was recently extended in [5] to any order in $\mathbb{Q}[\theta]$, but on very restricted classes of instances. In the present document, we extend the work of [5] to show how we can derive relations between ideals of norm bounded by $L_{\Delta}(a, c_1)$ in time $L_{\Delta}(b, c_2)$ in wider classes of number fields such that the specialization to $a = b = \frac{1}{3}$ corresponds to the result stated in [5].

Our $\mathfrak{q}$-descent method applies to some classes of orders $\mathcal{O}$ in number fields $K = \mathbb{Q}(\theta) = \mathbb{Q}[X]/T(X)$ for $\Delta = \operatorname{disc}(\mathcal{O})$. These classes are parametrized by constants $n_0, d_0 > 0$ and $0 < \alpha < \frac{1}{2}$. Let $T(X) = t_n X^n + t_{n-1} X^{n-1} + \ldots + t_0 \in \mathbb{Z}[X]$, $n := [K : \mathbb{Q}]$ and d be a bound on the size of the coefficients of T, that is $d := \log H_T$, where $H_T := \max_i |t_i|$. We say that

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

the order $\mathcal{O}$ belongs to $\mathcal{C}_{n_0,d_0,\alpha}$ if its discriminant Δ satisfies

$$n = n_0 \log(|\Delta|)^{\alpha}(1 + o(1)) \tag{3.3}$$
$$d = d_0 \log(|\Delta|)^{1-\alpha}(1 + o(1)). \tag{3.4}$$

In the rest of the paper, we will use $\kappa := n_0 d_0$ in the expression of the complexities.

These conditions on n and d are necessary because our approach is based on the number field sieve whose performances depend on the properties of the defining polynomial of the number field. However, it is easy to exhibit classes of orders satisfying conditions (3.3) and (3.4). For example, as shown in [**4**, § 2], the equation orders of number fields defined by $X^n - k$ where $k := \lfloor e^{\log(|D|)^{1-\alpha}} \rfloor$ and $n := \lfloor \log(|D|)^{\alpha} \rfloor$ are in $\mathcal{C}_{1,1,\alpha}$. These classes were described as a proof of concept. In [**5**, § 4.4], Biasse describes a potential application of this $\mathfrak{q}$-descent method to the evaluation of isogenies between Abelian varieties over a finite field of cardinality q with complex multiplication and the computation of their endomorphism ring. In this case, conditions (3.3) and (3.4) are equivalent to a condition of the form $g \sim \log(q)^{\delta}$ for some $\delta > 0$, where g is the dimension of the Abelian variety. The method of [**5**, § 4.4] allows the computation of relations between ideals in the polarized class group $\mathfrak{C}(\mathcal{O})$, in subexponential time, but the actual computation of isogenies requires the l-torsion, which is too expensive with the current methods to generalize the work of Jao–Soukharev [**20**] (on isogeny computation) and Bisson [**8**] (on endomorphism ring computation).

First decomposition. A modified version of Algorithm 3 needs to be called at the beginning of the $\mathfrak{q}$-descent to write the class $[\mathfrak{a}]$ of the ideal $\mathfrak{a}$ in $\mathrm{Cl}(\mathcal{O})$ as a power-product of classes of degree 1 prime ideals of norm bounded by $|\Delta|$. This restriction on the degree allows us to search for short elements of a simple form in these primes, as we see in the proof of Theorem 3.1. We describe the first decomposition in Algorithm 4 and analyze it in Lemma 3.3.

Lemma 3.3 (GRH + Heuristic 2). *Let $\mathfrak{a}$ be an ideal in an order $\mathcal{O}$ of a number field K in $\mathcal{C}_{n_0,d_0,\alpha}$ for some n_0, d_0, α with $0 < \alpha < \frac{1}{2}$. then we can write $[\mathfrak{a}]$ as a power-product of classes of prime ideals of the form*

$$\mathfrak{q} = q\mathcal{O} + (\theta - v_q)\mathcal{O},$$

where $v_q \in \mathbb{Z}$ and $\mathcal{N}(\mathfrak{q}) \leqslant |\Delta|$, in complexity $\log(\mathcal{N}(\mathfrak{a}))^{1+o(1)} L_{\Delta}(b, o(1))$ for any $0 < b \leqslant 1$.

Proof. We apply the same procedure as in Algorithm 3 with the parameter $k = \log(|\Delta|)^{b-\varepsilon}$ for an arbitrarily small $\varepsilon > 0$. This way, the reduced ideals $\mathfrak{b}$ satisfy $\mathcal{N}(\mathfrak{b}) \leqslant |\Delta|^{O(1)}$ and are derived in time $L_{\Delta}(b, o(1))$. According to Corollary 3.1, only $L_{\Delta}(b, o(1))$ of them need to be drawn until one which is $|\Delta|$-smooth is found, and under Heuristic 2, we can assume this decomposition to only include prime ideals of inertia degree 1. □

Algorithm 4. First decomposition.

Input: Ideal $\mathfrak{a}$, and $0 < b \leqslant 1$.
Output: Primes $\mathfrak{q}_i$ of inertia degree 1 and norm in $O(|\Delta|)$, and $\phi \in K$ such that $\mathfrak{a} = (\phi) \prod_i \mathfrak{q}_i$.

1: $k \leftarrow \log(|\Delta|)^{b-\varepsilon}$ for $0 < \varepsilon < b$.
2: $\mathfrak{a} \leftarrow (\phi_1)\mathfrak{a}$ where ϕ_1 is the output of Algorithm 2 on $(\mathfrak{a}, k)$.
3: found $\leftarrow$ false
4: **while** found = false **do**
5: $\quad (\mathfrak{q}_i)_{i \leqslant N} \leftarrow$ random degree 1 prime ideals with $\mathcal{N}(\mathfrak{q}_i) \leqslant 12 \log(|\Delta|)^2$.
6: $\quad \mathfrak{a}' \leftarrow (\phi_i) \cdot \mathfrak{a} \prod_i \mathfrak{q}_i^{-1}$ where ϕ_2 is the output of Algorithm 2 on $(\mathfrak{a} \cdot \prod_i \mathfrak{q}_i^{-1}, k)$.
7: $\quad$ **if** $\mathfrak{a}'$ is $|\Delta|$-smooth with respect to the prime ideals of inertia degree 1 **then**

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

found ← true
Let β and $(\mathfrak{p}_j)$ such that $\mathfrak{a}' = (\beta)\prod_j \mathfrak{p}_j$.
end if
end while
return $\{(\mathfrak{q}_i)_{i\leqslant N}, (\mathfrak{p}_j)\}$, $\phi_1 \cdot \phi_2 \cdot \beta$

The recursive decomposition. Once the first decomposition is done, the prime ideals occurring in the decomposition of $\mathfrak{a}$ are recursively decomposed as power-products of prime ideals of lower norm. For that, we enumerate elements $\phi \in \mathfrak{q} = q\mathcal{O} + (\theta - v_q)\mathcal{O}$ of the form $\phi = A(\theta)$ until one is smooth.

THEOREM 3.1 (GRH + Heuristic 2). *Let $\mathcal{O}$ be an order of a number field $K = \mathbb{Q}(\theta) \in \mathcal{C}_{n_0,d_0,\alpha}$ for $0 < \alpha < \frac{1}{2}$. We can find a B-smooth ideal equivalent to a $|\Delta|$-smooth ideal $\mathfrak{a} \subseteq \mathcal{O}$ with a decomposition in degree 1 prime ideals in time $L_\Delta(b, \mu)$, for some $\mu > 0$ where $B = L_\Delta(a, \rho)$ for some $\rho > 0$ satisfying:*

$$\text{(i) } 1 - b \geqslant \alpha \geqslant 1 - a - b; \quad \text{(ii) } b + 2a \geqslant 1; \quad \text{(iii) } 2b + a \geqslant 1; \quad \text{(iv) } b \geqslant \alpha.$$

Proof. We know that $\mathfrak{a}$ splits into the prime ideals of the form $\mathfrak{q} = q\mathcal{O} + (\theta - v_q)\mathcal{O}$ and of norm bounded in $O(|\Delta|)$. We proceed recursively, starting from the primes of the first decomposition. At each stage, $\mathfrak{q} = q\mathcal{O} + (\theta - v)\mathcal{O}$ is an ideal of norm bounded by $L_\Delta(a + \tau, c)$ for some $c > 0$ and $0 \leqslant \tau \leqslant 1 - a$. At the beginning we have $\tau = 1 - a$ and $c = 1$. We search for $\phi \in \mathfrak{q}$ such that $(\phi)/\mathfrak{q}$ is $L_\Delta(a + \tau/2, c')$-smooth for a c' depending on c. Such a ϕ satisfies $\mathfrak{q} \mid (\phi)$ and thus $[\mathfrak{q}]$ can be decomposed as a power-product of classes of prime ideals involved in the decomposition of (ϕ). We repeat this process until we obtain a decomposition involving only classes of elements of $\mathcal{B}$.

At each stage, we consider the ϕ belonging to the lattice of polynomials in θ of degree bounded by

$$k := \left\lfloor \sigma \frac{n}{(\log|\Delta|/\log\log|\Delta|)^{1-\beta-\tau/2}} \right\rfloor,$$

where $\sigma > 0$ is a constant to be determined later and $\beta := a + b$. Note that condition (i) implies that $k \to \infty$. When $\beta + (\tau/2) < 1$, we have $k < n$, and when $\beta + (\tau/2) \geqslant 1$, we set $k = \sigma n^{1-\varepsilon}$ for $\varepsilon > 0$ arbitrarily small. These ϕ form a $\mathbb{Z}$-lattice generated by

$$(v_0, \theta - v_1, \ldots, \theta^k - v_k),$$

with $v_0 = q$ and $v_i = v_q^i \mod q$ for $i \geqslant 1$. We want to spend the same time $L_\Delta(b, e + o(1))$ at each smoothing step for $e > 0$ to be optimized later. The search space has to be of the same size. We thus look for $L_\Delta(b, e + o(1))$ distinct $(k + 1)$-tuples $(\alpha_1, \ldots, \alpha_{k+1}) \in \mathbb{Z}^{k+1}$. Using Lemma 3.5, for every integer z, we can find e^{kz} such tuples satisfying $\log|\alpha_i| \leqslant D/k + z$ for $i \leqslant k + 1$ and $\log|\sum_i \alpha_i v_i| \leqslant D/k + z$. We adjust the value of z to make sure that all the $L_\Delta(b, e + o(1))$ tuples obtained during the search phase satisfy this property by solving $e^{kz} = L_\Delta(b, e + o(1))$. This yields

$$\begin{aligned} z &= \frac{e}{\sigma}\frac{1}{n}\log(|\Delta|)^b \log\log(|\Delta|)^{1-b}\left(\frac{\log(|\Delta|)}{\log\log(|\Delta|)}\right)^{1-\beta-\tau/2} \\ &= \frac{e}{\sigma}\frac{1}{n}\log(|\Delta|)^{1+b-\beta-\tau/2}\log\log(|\Delta|)^{1-(1+b-\beta-\tau/2)} \\ &= \frac{e}{\sigma}\frac{1}{n}\log(|\Delta|)^{1-a-\tau/2}\log\log(|\Delta|)^{1-(1-a-\tau/2)} = \frac{1}{n}\log L_\Delta(1 - a - \tau/2, e/\sigma + o(1)). \end{aligned}$$

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

Note that condition (ii) implies that $1-a \leqslant \beta$. Let $D = c\log(|\Delta|)^{a+\tau}\log\log(|\Delta|)^{1-(a+\tau)} = \log(q)$. From [4, Lemma 2], $\log(\mathcal{N}(\phi)) \leqslant n(D/k+z) + dk + d\log(k) + k\log(n)$. Let us bound the different terms of this expression. First we have

$$n(D/k+z) \leqslant nc\log(|\Delta|)^{a+\tau}\log\log(|\Delta|)^{1-(a+\tau)}\left(\frac{\log(|\Delta|)}{\log\log(|\Delta|)}\right)^{1-\beta-\tau/2}\frac{1}{\sigma n} + \frac{e}{\sigma}\log(|\Delta|)^{\beta+\tau/2}\log\log(|\Delta|)^{1-(\beta+\tau/2)}.$$

Under condition (iii), we get $a+\tau+1-\beta-\tau/2 \leqslant \beta+\tau/2$, and therefore

$$n(D/k+z) \leqslant \frac{c}{\sigma}\log(|\Delta|)^{\beta+\tau/2}\log\log(|\Delta|)^{1-(\beta+\tau/2)} + \frac{e}{\sigma}\log(|\Delta|)^{\beta+\tau/2}\log\log(|\Delta|)^{1-(\beta+\tau/2)}.$$

Moreover, $kd = \sigma\kappa\log(|\Delta|)^{\beta+\sigma/2}\log\log(|\Delta|)^{1-(\beta+\tau/2)}$, so the algebraic integers ϕ we draw satisfy

$$\mathcal{N}(\phi) \leqslant L_\Delta(\beta+\tau/2, (c+e)/\sigma + o(1)).$$

Let $\mathfrak{q}'$ be the ideal such that $(\phi) = \mathfrak{q}\cdot\mathfrak{q}'$. Its norm is also bounded: $\mathcal{N}(\mathfrak{q}') \leqslant L_\Delta(\beta+\tau/2, (c+e)/\sigma + o(1))$. Note that this condition also holds when $\beta+\tau/2 \geqslant 1$ and $k = n^{1-\varepsilon}$. From Heuristic 2 and Corollary 3.1, an $L_\Delta(a+\tau/2, c')$-smooth decomposition is found in time $L_\Delta(b,e)$ if c' satisfies

$$c' = \frac{b}{e}\left(\frac{c+e}{\sigma} + \sigma\kappa\right).$$

The optimal σ is $\sqrt{(c+e)/\kappa}$, so $c' = (2b/e)\sqrt{\kappa}\sqrt{c+e}$. In time $L_\Delta(b, e+o(1))$, we obtain an $L_\Delta(a+\tau/2^i, c_i)$-smooth decomposition where

$$c_i = \frac{2b}{e}\sqrt{\kappa}\sqrt{c_{i-1}+e}, \quad c_0 = 1.$$

Let $\chi = 2b\sqrt{\kappa}/e$, then $c_i = \chi\sqrt{c_{i-1}+e}$ and the limit c_∞ of $(c_i)_{i\in\mathbb{N}}$ satisfies $c_\infty = \chi\sqrt{c_\infty+e}$, and the positive solution to this equation is

$$c_\infty = \frac{\chi}{2}(\chi + \sqrt{\chi^2+4e}).$$

To estimate the number of steps necessary to reach an $L_\Delta(a,\rho)$-smooth decomposition for some $\rho > 0$, notice that $L_\Delta(a+\tau/2^{i-1}, c_i) = L_\Delta(a, c_i\cdot\mathcal{M}^{\tau/2^{i-1}})$ for $\mathcal{M} = \log(|\Delta|)/\log\log(|\Delta|)$. Let $\xi > 0$ be an arbitrary constant. After a number of steps only depending on e, κ and ξ, we have $c_i < c_\infty(1+\xi)$, and after $O(\log\log|\Delta|)$ steps $\mathcal{M}^{\tau/2^{i-1}} < (1+\xi)$. We can thus decompose $[\mathfrak{a}']$ as a power-product of classes of prime ideals of norm bounded by

$$L_\Delta(a, c_\infty(1+\xi)).$$

At each step, the decomposition involves $O(\log|\Delta|)$ ideals, and we perform $O(\log\log|\Delta|)$ steps. Therefore we obtain an $L_\Delta(b,\rho)$-smooth decomposition for $\rho = c_\infty(1+\xi)$ in time $L_\Delta(b, e+o(1))$ and there are $L_\Delta(z, o(1))$ ideals in the final decomposition. □

Algorithm 5. $\mathfrak{q}$-descent.

Input: Ideal $\mathfrak{a}$, $\mathcal{B} = \{\mathfrak{p}_1, \ldots, \mathfrak{p}_N\}$.

Output: Primes $(\mathfrak{q}_i)_{i\leqslant l} \in \mathcal{B}$, integers (e_i) and $(\phi_j)_{j\leqslant k} \in K$ such that $\mathfrak{a} = \prod_{j\leqslant k}(\phi_j)\cdot\prod_{i\leqslant l}\mathfrak{q}_i^{e_i}$.

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

1: Find prime ideals $(\mathfrak{q}_i)_{i\leqslant l}$ of norm bounded by $|\Delta|$ and ϕ_1 with $\mathfrak{a} = (\phi)\cdot\prod_i \mathfrak{q}_i$ using Algorithm 4.
2: genList $\leftarrow \{\phi_1\}$, primeList $\leftarrow \{\mathfrak{q}_1,\dots,\mathfrak{q}_l\}$, expList $\leftarrow \{1,\dots,1\}$.
3: **while** there is $\mathfrak{q}\notin\mathcal{B}$ in the decomposition of $[\mathfrak{a}]$ **do**
4: Find $(\mathfrak{q}_i)_{i\leqslant l}$, $(e_i)_{i\leqslant l}$ and ϕ_k such that $\mathfrak{q} = (\phi_k)\prod_{i\leqslant l}\mathfrak{q}_i^{e_i}$ as in Theorem 3.1.
5: genList $\leftarrow$ genList $\cup\{\phi_k\}$, primeList $\leftarrow$ primeList $\cup\{\mathfrak{q}_1,\dots,\mathfrak{q}_l\}$.
6: expList $\leftarrow \{e_1,\dots,e_l\}$.
7: **end while**
8: **return** genList, primeList, expList.

COROLLARY 3.4. *We can find an $L_\Delta(a,c_1)$-smooth decomposition of a $|\Delta|$-smooth ideal of $\mathcal{O}$ in $K\in\mathcal{C}_{n_0,d_0,\alpha}$ in time $L_\Delta(a,c_2)$ if a satisfies $1-a\geqslant\alpha\geqslant 1-2a$, $a\geqslant\frac{1}{3}$, and $a\geqslant\alpha$.*

Example.
- We can achieve an $L_\Delta(\frac{2}{5},c_1)$-smooth decomposition in time $L_\Delta(\frac{2}{5},c_2)$ for some $c_1,c_2>0$ in orders of $K\in\mathcal{C}_{n_0,d_0,\alpha}$ where $\frac{2}{5}\geqslant\alpha\geqslant\frac{1}{5}$.
- We can achieve an $L_\Delta(\frac{1}{3},c_1)$-smooth decomposition in time $L_\Delta(\frac{1}{3},c_2)$ for some $c_1,c_2>0$ in orders of $K\in\mathcal{C}_{n_0,d_0,\alpha}$ for $\alpha=\frac{1}{3}$, thus recovering the result of [5].

Finding short elements in $\mathfrak{q}$. We draw elements $\phi\in\mathfrak{q} = q\mathcal{O}+(\theta-v_q)\mathcal{O}$ in the $\mathbb{Z}$-lattice generated by $(v_0,\theta-v_1,\dots,\theta^k-v_k)$ where $v_0=q$ and $v_i = v_0^i \bmod q$. In the proof of Theorem 3.1 we rely on the fact that there are sufficiently many small elements in a bounded hypercube of this lattice.

LEMMA 3.5. *Let $\sigma,\tau,c,a,\beta>0$ be as defined in the proof of Theorem 3.1 and some integers D and k be defined by*

$$k := \left\lfloor \sigma\frac{n}{(\log|\Delta|/\log\log|\Delta|)^{1-\beta-\tau/2}}\right\rfloor,\quad D := \log(L_\Delta(a+\tau,c)).$$

Let $v_1,\dots,v_{k+1}$ be integers satisfying $\log|v_i|\leqslant D$. Then, for any integer z, there exist at least e^{kz} tuples $(\alpha_1,\dots,\alpha_{k+1})\in\mathbb{Z}^{k+1}$ satisfying

$$\log|\alpha_i|\leqslant D/k+z \quad\text{and}\quad \log\left|\sum_i\alpha_i v_i\right|\leqslant D/k+z.$$

Proof. Let us define the $k+1$ dimensional lattice Λ generated by the rows of

$$A := \begin{pmatrix} 1 & 0 & \dots & 0 & v_1\\ 0 & 1 & \ddots & \vdots & \vdots\\ \vdots & \ddots & \ddots & 0 & \vdots\\ 0 & \dots & 0 & 1 & v_{k+1}\end{pmatrix}.$$

For any element $x\in\Lambda$, there exists $(\alpha_1,\dots,\alpha_{k+1})\in\mathbb{Z}^{k+1}$ such that

$$x = \left(\alpha_1,\dots,\alpha_{k+1},\sum_i\alpha_i v_i\right).$$

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

The determinant $d(\Lambda)$ of Λ satisfies

$$d(\Lambda) = \sqrt{\det(AA^T)} = \sqrt{\sum_{i\leqslant k+1} v_i + \sum_{i\leqslant k+1} v_i v_{k+1-i}} \leqslant (\sqrt{2k+1})e^D.$$

Let $X \subset \mathbb{R}^{k+2}$ be the symmetric and convex set of points defined by

$$X = \{(x_1, \ldots, x_{k+2}) \mid \text{for all } i\ |x_i| \leqslant D/k + z\}.$$

The volume $V(X)$ equals $2^{k+2}e^{(k+2)(D/k+z)}$, and from [11, III.2.2, Theorem II] we know that if $V(X) > m2^{k+2}d(\Lambda)$, then X intersects Λ in at least m pairs of points $\pm x \in \mathbb{R}^{k+2}$. It thus suffices to prove that

$$e^{kz} < \frac{e^{(k+2)(D/k+z)}}{e^D\sqrt{2k+1}} = e^{kz} \cdot \frac{e^{2D/k+2z}}{\sqrt{2k+1}},$$

which is satisfied since $D/k = (c/\sigma)\log|\Delta|^{2/3-\alpha+\tau/2}\log\log|\Delta|^{1/3-\tau/2} \gg \log(2k+1)$. □

These short vectors need to be found via an enumeration algorithm of short vectors. This is exponential in the dimension of the lattice, which itself is bounded by n. We use the method described in [19] to perform this search.

PROPOSITION 3.6. *The search for the solution of the restrictions described in Lemma 3.5 takes time bounded by $L_\Delta(b, e + o(1))$ if condition* (iv) *of Theorem 3.1 is satisfied.*

Proof. Enumerating vectors of length bounded by $A = e^{D/k+z}$ with [19, Algorithm 10] takes

$$2^{O(k)}\frac{A^k}{k^{k/2}d(\Lambda)} \leqslant 2^{O(k)}\frac{e^{D+kz}}{k^{k/2}e^D} \leqslant 2^{O(k)}L_\Delta(a, e + o(1)).$$

Furthermore, since $k \sim \log(|\Delta|)^{\alpha-\varepsilon}$ for some $\varepsilon > 0$, we have $2^{O(k)} = L_\Delta(b, o(1))$ from condition (iv), which terminates the proof. □

4. *Finding the class group and the unit group from relations*

Let $\mathcal{B} = \{\mathfrak{p}_1, \ldots, \mathfrak{p}_N\}$ be a factor base of ideals whose classes generate $\mathrm{Cl}(\mathcal{O})$. Then the vectors $(e_1, \ldots, e_N)$ such that $\mathfrak{p}_1^{e_1} \ldots \mathfrak{p}_N^{e_N} = (\alpha)$ for some $\alpha \in \mathcal{O}$ form a $\mathbb{Z}$-module $\mathcal{L}$ in $\mathbb{Z}^N$. Once $\mathcal{L}$ has full rank, we compute a tentative class group and unit group. This is done by finding a basis of $\mathcal{L}$ together with kernel vectors from the Hermite normal form (HNF) of $\mathcal{L}$. Then the Smith normal form (SNF) of $\mathbb{Z}^N/\mathcal{L}$ gives us its group structure while we derive a minimal generating set of the units coming from kernel vectors of $\mathcal{L}$. Finally, the validity of the result is established by testing the product of $|\mathbb{Z}^N/\mathcal{L}|$ and the volume of the current lattice of logarithm vectors of units against the theoretical value of $|\mathrm{Cl}(\mathcal{O})|R$ given by an Euler product (where R is the regulator). In the following, we present these steps separately.

4.1. *Computing the tentative class group*

We assume that we are given a full rank $N' \times N$ relation matrix M as input satisfying $N \leqslant N' \in L_\Delta(a, c+o(1))$ for $0 < a < 1$, $c > 0$ and $|M| \leqslant |\Delta|$ where $|M|$ is the maximum of the absolute value of the coefficients of M. We use the HNF algorithm described in Storjohann's thesis [34, Chapter 6] which provides us with bounds on the coefficients of kernel vectors, which we need to compute units.

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

PROPOSITION 4.1. *If the relation matrix $M \in \mathbb{Z}^{N' \times N}$ satisfies $N \leqslant N' \in L_\Delta(a, c+o(1))$ for $0 < a < 1$ and $|M| \leqslant |\Delta|$, then we can find a tentative class group G and a kernel $\ker(M)$ in time $L_\Delta(a, (\nu+1)c + o(1))$ with:*

- $\log(\#G) \in L_\Delta(a, c + o(1))$;
- $\dim(\ker(M)) \in L_\Delta(a, c + o(1))$;
- $\log|\ker(M)| \in L_\Delta(a, c + o(1))$,

where $2 \leqslant \nu < 3$ is the exponent of matrix multiplication, $\#G$ is the cardinality of G and $|A|$ is the maximum absolute value of the entries of A.

Proof. We use the HNF and SNF algorithms from Storjohann's thesis, which do not require the matrix to be sparse. Note that in practical computations, a relation matrix is usually sparse. The computation of the Hermite form costs $O(N^\nu \log(\delta) + N^2 \log(N)\, \mathrm{B}(\log(\delta))$ where $\delta = (\sqrt{N}|M|)^N$ and $\mathrm{B}(x) = x \log^2(x) \log\log(x)$. Moreover, the coefficients of the resulting Hermite form H, as well as the coefficients of the matrix U such that $UM = H$, are bounded by δ. The computation of the nullspace and of the Smith form is dominated by the computation of the HNF, and the dimension of the nullspace is $N' - N$. The result is obtained by substituting the values of $|M| = |\Delta|$ and $N' \in L_\Delta(a, c + o(1))$ in the above formulas. □

Note that for the final class group we have $\log(\#G) \in L_\Delta(0, \frac{1}{2})$, however, we might not have enough relations yet.

4.2. *Computing the tentative unit group*

Assume we are given $k := N' - N \in L_\Delta(a, c + o(1))$ kernel vectors $X \in \mathbb{Z}^{N'}$ satisfying $\log(|X|) \in L_\Delta(a, c + o(1))$ together with the N' elements α_i such that

$$(\alpha_i) = \mathfrak{p}_1^{m_{i,1}} \dots \mathfrak{p}_N^{m_{i,N}},$$

where $M = (m_{i,j})_{i \leqslant N', j \leqslant N}$ is the relation matrix. Furthermore, let S be such that for all $i \leqslant N'$ and for all $j \leqslant r+1$, we have $\log|\alpha_i|_j \leqslant S$. We wish to compute a minimal generating set for the group U generated by the units of the form $\beta_X = \alpha_1^{x_1} \dots \alpha_{N'}^{x_{N'}}$, where $X = (x_i)_{i \leqslant N'}$ is a kernel vector. Let $r = r_1 + r_2$ where r_1 is the number of real embeddings and r_2 the number of complex embeddings. This boils down to finding r vectors that span

$$\mathcal{L}_\mathbb{R} := \{(\log|\beta_X|_1, \dots, \log|\beta_X|_{r+1}) \mid \beta_X \in U\}.$$

General strategy. Let A be the matrix given by

$$A = \left(\sum_{j \leqslant k} x_j \log|\alpha_j|_1, \dots, \sum_{j \leqslant k} x_j \log|\alpha_j|_{r+1} \right)_{i \leqslant k}.$$

Let A_r be a matrix formed with r independent rows in A. These span the $\mathbb{Q}$-vector space generated by the rows of A. Let $B \in \mathbb{Q}^{k \times r}$ be such that $BA_r = A$ and $Q_{\mathrm{com}} \in \mathbb{Z}_{>0}$ minimal such that $Q_{\mathrm{com}} B \in \mathbb{Z}^{k \times r}$. We compute the HNF H_B of $Q_{\mathrm{com}} B$, along with $U \in \mathrm{GL}_{k \times k}$ such that

$$U \cdot Q_{\mathrm{com}} B = \begin{pmatrix} H_B \\ \cdots\cdots\cdots \\ (0) \end{pmatrix}.$$

Let U_r be the first r rows of U. Then $U_r A$ span $\mathcal{L}_\mathbb{R}$ (over $\mathbb{Z}$). Indeed, it suffices to notice that $UA = UBA_r$ is of the form $(H_\mathbb{Q}/(0))$ for some $H_\mathbb{Q} \in \mathbb{Q}^{r \times (r+1)}$. Then any row of A arises as a $\mathbb{Z}$-linear combination of rows of $H_\mathbb{Q} = U_r A$.

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

A p-adic approach. In [4], the approach described above was carried out by using fixed point approximations of the $\log|\alpha_i|_j$. In [7], a heuristic p-adic approach was described and implemented. It greatly simplifies the estimation of the required precision and the analysis of the theoretical complexity (which was not done in [7]). Moreover, it allows the parallelization of the process. In this section, we propose a rigorous analysis of the computation of the tentative unit group with p-adic approximations. Our general strategy differs from that of [7] in the sense that we compute the generating set directly from all the kernel vectors instead of doing it iteratively.

We choose an unramified prime p such that the p-adic splitting field K_p has moderate degree $d \in O(n^{\frac{1}{2}\log n})$. Note that the degree of a splitting field K_p is the order of the Frobenius automorphism at p, hence the order of some element in the Galois group of the defining polynomial of K. By the famous theorem of Erdős and Turán [14], asymptotically every second element in $S(n)$ has order bounded by $\exp(\frac{1}{2}\log^2 n)$ hence every second prime will give rise to a p-adic splitting field of degree bounded by this. Furthermore, a 'random' field K is expected to have Galois group $S(n)$. In case the group is much smaller, the splitting field can also be chosen much smaller. As the splitting field $K_p = \mathbb{Q}_p[t]/g$ we choose $g \in \mathbb{Z}[t]$ as a monic lift of a defining polynomial for $\mathbb{F}_{p^d}$ the finite field with p^d elements. We are of course only working in approximations with a fixed precision, that is, in $R_p = (\mathbb{Z}_p/p^m)[t]/g = (\mathbb{Z}/p^m)[t]/g$. The costs of operations is thus $O(d\log d\log\log dm\log m\log p\log\log p) = L_\Delta(\varepsilon, c)$ for all $\varepsilon > 0$ ($n = n_0\log^\alpha\Delta(1+o(1))$, thus $d = \exp(\frac{1}{2}\log^2 n) = \exp(\frac{1}{2}\log^2(n_0\log^\alpha\Delta(1+o(1))))$). Then we have n embeddings ϕ_i of $K \to K_p$, and we define a map $L_p : K^* \to K_p^n : x \mapsto (\log\phi_i(x))_i$ where ϕ_i is the usual p-adic logarithm extended to K_p. Let m be the p-adic precision we work with and $A^{(p)} \in (\mathbb{Z}/p^m\mathbb{Z})^{k\times(r+1)}$ the p-adic approximation of the matrix A previously described. We first find r independent rows of $A^{(p)}$ to form $A_r^{(p)}$ (they exist assuming Leopoldt's conjecture). Then we solve the linear systems $XA_r^{(p)} = a_i$ in the p-adics and perform a rational reconstruction of the solutions. Provided that the p-adic precision is large enough, this yields $B \in \mathbb{Q}^{k\times r}$ previously defined, which allows us to compute our generating set.

We solve the linear systems $XA_r^{(p)} = \vec{a}_i$ by first precomputing the SNF of $(A_r^{(p)}/(0)) \in (\mathbb{Z}/p^m\mathbb{Z})^{(r+1)\times(r+1)}$ with the method described in [34, Chapter 7]. This yields two invertible matrices $U, V \in (\mathbb{Z}/p^m\mathbb{Z})^{r\times r}$ such that $U(A_r^{(p)}/(0))V = \mathrm{diag}(d_1,\ldots,d_r,0)$. Then each system $XA_r^{(p)} = \vec{a}_i$ is solved by first calculating $\vec{a}'_i = (\vec{a}_i \mid 0)V$, then solving $X'D = \vec{a}'_i$ where $D = \mathrm{diag}(d_1,\ldots,d_r,0)$ by performing r divisions in $(\mathbb{Z}/p^m\mathbb{Z})$, and finally returning the first r coefficients of $X = X'U$. This is correct because

$$X\begin{pmatrix}A_r^{(p)}\\(0)\end{pmatrix} = X'U\begin{pmatrix}A_r^{(p)}\\(0)\end{pmatrix} = X'U\begin{pmatrix}A_r^{(p)}\\(0)\end{pmatrix}VV^{-1} = X'DV^{-1} = \vec{a}'_iV^{-1} = (\vec{a}_i \mid 0).$$

This naturally extends to the resolution of linear systems of a rectangular matrix with less than r rows, and it also allows us to decide if such a system has a solution, which we use to extract r independent rows of $A^{(p)}$.

Algorithm 6. Tentative unit group.

Input: Generators $(\alpha_i)_{i\leqslant N'}$, kernel vectors $(o_i)_{i\leqslant k}$ of the relation matrix, p, m.

Output: $(V_i)_{i\leqslant r}$ in $\mathbb{Z}^{N'}$ such that $(\prod_j \alpha_j^{V_{i,j}})_{i\leqslant r}$ generates all the units derived from $(o_i)_{i\leqslant k}$.

1: Compute $G \in (\mathbb{Z}/p^m\mathbb{Z})^{N'\times r}$ whose row vectors are the p-adic approximations of $(\log|\alpha_i|_1,\ldots,\log|\alpha_i|_r)$.

2: $A^{(p)} \leftarrow (o_i)_iG$, $A_r^{(p)} \leftarrow A_{(1)}^{(p)}$, $i \leftarrow 2$, where $A_{(i)}$ denotes the ith row of A.

3: **while** $A_r^{(p)}$ has less than r rows **do**

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

4: If $XA_r^{(p)} = A_{(i)}^{(p)}$ has no solution, $A_r^{(p)} \leftarrow \begin{pmatrix} A_r^{(p)} \\ A_{(i)}^{(p)} \end{pmatrix}$.
5: $i \leftarrow i + 1$
6: **end while**
7: $B \leftarrow \{\}$. Precompute the Smith form of $\begin{pmatrix} A_r^{(p)} \\ (0) \end{pmatrix}$.
8: **for** $i \leqslant k$ **do**
9: $b \leftarrow A_{(i)}^{(p)}$. Solve $X' A_r^{(p)} = b$.
10: Let $X \in \mathbb{Q}^r$ be the rational reconstruction of X'.
11: $B \leftarrow \begin{pmatrix} B \\ X \end{pmatrix}$.
12: **end for**
13: Let Q_{com} the common denominator of B and U such that $U \cdot Q_{\mathrm{com}} B = \mathrm{HNF}(Q_{\mathrm{com}} B)$.
14: Let U_r be the first r rows of U and $V = U(o_i)_i$.
15: **return** The row vectors $(V_i)_{i \leqslant r}$ of V.

PROPOSITION 4.2. *The complexity of Algorithm 6 is in*

$$O(L_\Delta(a, c + o(1)) \cdot \mathrm{B}(\log(rp^m))),$$

where $\mathrm{B}(x) = x(\log(x))^2 \log\log(x)$, *$m$ is the chosen p-adic precision, and $L_\Delta(a, c)$ is a bound on the bit size of the entries of the kernel vectors $(K_i)_{i \leqslant k}$. Furthermore, the entries of the output V satisfy*

$$\log |V| \in O(L_\Delta(a, c + o(1)) + \log(rp^m)).$$

Proof. To solve our linear systems in $(\mathbb{Z}/p^m\mathbb{Z})^{r \times r}$, we need to compute the Smith form of an $(r+1) \times (r+1)$ matrix. If the row dimension of the matrix is lower than $r+1$, we complete it with rows of 0. We use the SNF algorithm described in [**34**, Chapter 7]. Its complexity is in

$$O(r^\nu \log(r) \log(rp^m) + r^2 \log(r)\, \mathrm{B}(\log(rp^m))).$$

Between Steps 3 to 6 of Algorithm 6, we compute a maximum of $L_\Delta(a, c + o(1))$ Smith forms of $(r+1) \times (r+1)$ matrices. Then one more is required in Step 7, before the series of k linear system resolutions of Steps 8–12. The multiplications $b \leftarrow (A_{(i)}^{(p)} \mid 0)V$ and $X' \leftarrow X'U$ cost $O(r^2\, \mathrm{B}(\log(rp^m)))$ while the r divisions leading to the solution of $X'D = b$ cost $O(r\, \mathrm{B}(\log(rp^m)))$.

Finally, one HNF computation needs to be performed in Step 13 of Algorithm 6. The matrix $Q_{\mathrm{com}} B$ is in $\mathbb{Z}^{k \times r}$, and the bit size of its entries is bounded by $O(\log(p^m))$ since they are the result of a rational reconstruction from elements in $(\mathbb{Z}/p^m\mathbb{Z})$. Therefore according to [**34**, Proposition 6.3], this costs

$$O(kr^{\nu-1} \log(\delta) + kr \log(r)\, \mathrm{B}(\log(\Delta))),$$

where $\delta = (\sqrt{r} p^m)^r$. Note that $k \in L_\Delta(a, c + o(1))$ while $r \in L_\Delta(a, o(1))$, so this complexity is bounded by the one stated in the claim. We also know that $|U| \leqslant (\sqrt{r} p^m)^r$, therefore, $|V| \leqslant k |(o_i)_i| (\sqrt{r} p^m)^r$, and the result follows from the fact that $\log |o_i| \in L_\Delta(a, c + o(1))$. □

The last part that needs to be addressed is the choice of the p-adic precision m. It directly derives from the essential requirement for the rational reconstruction of a rational number d/l, namely that $p^m \geqslant 2ld$.

PROPOSITION 4.3. *Let S be such that $\log |\alpha_i|_j \leqslant S$ for $i \leqslant N', j \leqslant r$ and a, c such that the bit size of the entries of the kernel vectors and the number N' of relations collected are in $L_\Delta(a, c + o(1))$, then the p-adic precision must satisfy*

$$m \in O(r \log(S) + L_\Delta(a, c + o(1))).$$

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

Proof. We assume that each α_i coming from the relation search satisfies $\log|\alpha_i|_j \leqslant S$. Therefore, each $\beta_{X_i} = \alpha_1^{x_{i,1}} \dots \alpha_{N'}^{x_{i,N'}}$ coming from a kernel vector X_i satisfies $\log|\beta_{X_i}|_j \leqslant SN'|X| \in S \cdot 2^{L_\Delta(a,c+o(1))}$. Let $L(x) := (\log|x|_1, \dots, \log|x|_r)$. Then, each solution in Step 10 of Algorithm 6 is a solution to an equation of the form $\sum_{j\leqslant r} x_j L(\beta_{X_j}) = L(\beta_{X_{j_0}})$. The $x_j \in \mathbb{Q}$ can be expressed as

$$x_j = \det(L(\beta_{X_1}), \dots, L(\beta_{X_{j_0}}), \dots, L(\beta_{X_r}))/\det(L(\beta_{X_1}), \dots, L(\beta_{X_r})).$$

The numerator and denominator of the above expression are bounded from below by 0.2 (an absolute lower bound on the regulator), and by the Hadamard bound $r^{r/2}(S \cdot 2^{L_\Delta(a,c+o(1))})^r$ from above. Therefore, we must have $p^m \geqslant 2r^r(S \cdot 2^{L_\Delta(a,c+o(1))})^{2r}$. □

In order to find the initial independent units, we utilize the same proof. The only difference is that, since we do not have a full rank system, we cannot use the lower regulator bound to obtain a lower bound for the denominator above. We have to replace this by a bound derived from the lower bound of the size of a non-torsion unit: $S(\beta) \geqslant \frac{21}{128}(\log n/n^2)$.

We note that an additional advantage of the p-adic approach is the inherent potential to parallelize: suppose after choosing independent units we then perform the above algorithm for two different p-adic splitting fields $K_{p_{1,2}}$ at *half* the precision each. The Chinese remainder theorem (CRT) will allow us to combine the relations.

4.3. *Certifying the result*

Assume we are given the elementary divisors of a tentative class group of cardinality h', the generators $(\alpha_i)_{i\leqslant N'}$ of the principal ideals of the relations and the row vectors $(V_i)_{i\leqslant r}$ in $Z^{N'}$ such that the $\prod_j \alpha_i^{V_{i,j}}$ are a minimal generating set of the units found from the kernel vectors of the relation matrix. If $\mathcal{O}$ is the maximal order, we use an approximation of the Euler product to certify the result: $hR = (|\mu|\sqrt{|\Delta|}/2^{r_1}(2\pi)^{r_2}) \lim_{s\to 1}((s-1)\zeta_K(s))$, where $\zeta_K(s) = \sum_{\mathfrak{a}} (1/\mathcal{N}(\mathfrak{a})^s)$ is the usual ζ-function associated to K and $|\mu|$ is the cardinality of μ the group of torsion units. Indeed, it allows us to derive a bound h^* in polynomial time under GRH that satisfies $h^* \leqslant hR < 2h^*$ (see [3]). If the tentative class number and regulator do not satisfy this inequality, we declare a failure (or in practice, collect more relations).

We want to adapt this to general orders. Let $\mathfrak{f}$ be the conductor of $\mathcal{O}$ in $\mathcal{O}_K$. Jenner [21] showed that

$$\zeta_{\mathcal{O}}(s) := \prod \left(1 - \frac{1}{\mathcal{N}_{\mathcal{O}}(\mathfrak{P})^{-s}}\right)$$

where the product runs over all maximal ideals in $\mathcal{O}$, and defines a zeta function for $\mathcal{O}$, that is a complex function with a meromorphic continuation on $\mathbb{C}$ with a simple pole at 1. Here, $\mathcal{N}_{\mathcal{O}}(A) = |\mathcal{O}/A|$ denotes the norm of some ideal A in $\mathcal{O}$. Defining $\phi_{\mathcal{O}}(s) := \prod(1 - \mathcal{N}_{\mathcal{O}}(\mathfrak{P})^{-s})$ for any order $\mathcal{O}$ and $s \in \mathbb{C}$ and the product running over all primes $\mathcal{O} \supset \mathfrak{P} \supseteq \mathfrak{f}$, he showed

$$\zeta_{\mathcal{O}}(s) = \zeta_{\mathcal{O}_K}(s) \frac{\prod(1 - \mathcal{N}_{\mathcal{O}_K}(\mathfrak{P})^{-s})}{\prod(1 - \mathcal{N}_{\mathcal{O}}(\mathfrak{P})^{-s})} = \zeta_{\mathcal{O}_K}(s) \frac{\phi_{\mathcal{O}_K}(s)}{\phi_{\mathcal{O}}(s)}$$

for the ordinary zeta function $\zeta_K = \zeta_{\mathcal{O}_K}$ of K and complex functions $\phi_{\mathcal{O}_K}, \phi_{\mathcal{O}}$.

Neukirch [26, Theorem I.12.12] shows that the index of the unit group of $\mathcal{O}$ in the unit group of $\mathcal{O}_K$ satisfies

$$(\mathcal{O}_K{}^* : \mathcal{O}^*) = \frac{h(\mathcal{O}_K)}{h(\mathcal{O})} \frac{|(\mathcal{O}_K/\mathfrak{f})^*|}{|(\mathcal{O}/\mathfrak{f})^*|}.$$

Furthermore [26, Theorem I.12.11], the Chinese remainder theorem yields

$$\mathcal{O}/\mathfrak{f} = \oplus \mathcal{O}_{\mathfrak{P}}/\mathfrak{f}\mathcal{O}_{\mathfrak{P}}$$

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

where $\mathcal{O}_{\mathfrak{P}}$ is the localization at $\mathfrak{P}$ and the sum runs over all maximal ideals $\mathfrak{P} \supseteq \mathfrak{f}$ (the terms are trivial for the others). Now we define the canonical projection

$$\Pi : \mathcal{O}_{\mathfrak{P}}/\mathfrak{f}\mathcal{O}_{\mathfrak{P}} \to \mathcal{O}_{\mathfrak{P}}/\mathfrak{P}\mathcal{O}_{\mathfrak{P}} = \mathcal{O}/\mathfrak{P}.$$

Since the map is surjective and all rings are finite, we get

$$|\mathcal{O}/\mathfrak{P}||\ker \Pi| = |\mathcal{O}_{\mathfrak{P}}/\mathfrak{f}\mathcal{O}_{\mathfrak{P}}|.$$

It is trivial to see that $x \in \mathcal{O}_{\mathfrak{P}}/\mathfrak{f}$ is a unit if and only if $\Pi(x)$ is a unit, so

$$|(\mathcal{O}_{\mathfrak{P}}/\mathfrak{f})^*| = (\mathcal{N}_{\mathcal{O}}(\mathfrak{P}) - 1)|\ker \Pi| = (\mathcal{N}_{\mathcal{O}}(\mathfrak{P}) - 1)\frac{|\mathcal{O}_{\mathfrak{P}}/\mathfrak{f}|}{\mathcal{N}_{\mathcal{O}}(\mathfrak{P})}.$$

Combining this with the CRT, we see

$$\begin{aligned}|(\mathcal{O}/\mathfrak{f})^*| &= \prod |(\mathcal{O}_{\mathfrak{P}}/\mathfrak{f})^*| = \prod \frac{\mathcal{N}_{\mathcal{O}}(\mathfrak{P}) - 1}{\mathcal{N}_{\mathcal{O}}(\mathfrak{P})} \prod |\mathcal{O}_{\mathfrak{P}}/\mathfrak{f}| = \mathcal{N}_{\mathcal{O}}(\mathfrak{f}) \prod \frac{\mathcal{N}_{\mathcal{O}}(\mathfrak{P}) - 1}{\mathcal{N}_{\mathcal{O}}(\mathfrak{P})} \\ &= \mathcal{N}_{\mathcal{O}}(\mathfrak{f})\phi_{\mathcal{O}}(1).\end{aligned}$$

Lastly, we note

$$(\mathcal{O}_K{}^* : \mathcal{O}^*) = (\mathcal{O}_K{}^* : \langle \mu(\mathcal{O}_K), \mathcal{O}^* \rangle)(\langle \mu(\mathcal{O}_K), \mathcal{O}^* \rangle : \mathcal{O}^*) = \frac{R(\mathcal{O})}{|\mu(\mathcal{O})|} \frac{|\mu(\mathcal{O}_K)|}{R(\mathcal{O}_K)}$$

where $\mu(\mathcal{O})$ denotes the torsion part of $\mathcal{O}^*$ and $|\mu(\mathcal{O})|$ is the size of the group it generates, that is, the order of $\mu(\mathcal{O})$.

Now we have all ingredients to get the class number formula for $\mathcal{O}$:

$$\begin{aligned}\operatorname{res} \zeta_{\mathcal{O}}|_{s=1} &= \operatorname{res} \zeta_{\mathcal{O}_K}|_{s=1} \frac{\phi_{\mathcal{O}_K}(1)}{\phi_{\mathcal{O}}(1)} \\ &= 2^{r_1}(2\pi)^{r_2} h(\mathcal{O}_K) \frac{R(\mathcal{O}_K)}{|\mu(\mathcal{O}_K)|} |\operatorname{disc}(\mathcal{O}_K)|^{-1/2} \frac{|\mathcal{O}_K/\mathfrak{f}|}{\mathcal{N}_{\mathcal{O}_K}(\mathfrak{f})} \frac{\mathcal{N}_{\mathcal{O}}(\mathfrak{f})}{|\mathcal{O}/\mathfrak{f}|} \\ &= 2^{r_1}(2\pi)^{r_2} (\mathcal{O}_K{}^* : \mathcal{O}^*) h(\mathcal{O}) \frac{\mathcal{N}_{\mathcal{O}}(\mathfrak{f})}{\mathcal{N}_{\mathcal{O}_K}(\mathfrak{f})} \frac{R(\mathcal{O}_K)}{|\mu(\mathcal{O}_K)|} |\operatorname{disc}(\mathcal{O}_K)|^{-1/2} \\ &= 2^{r_1}(2\pi)^{r_2} \frac{R(\mathcal{O}) h(\mathcal{O})}{|\mu(\mathcal{O})|} \mathcal{N}_{\mathcal{O}}^{-1}(\mathfrak{f}) |\operatorname{disc}(\mathcal{O}_K)|^{-1/2}.\end{aligned}$$

Since $\mathcal{N}_{\mathcal{O}}^2(\mathfrak{f}) = \mathcal{N}_{\mathcal{O}_K}(\mathfrak{f})$ and $\operatorname{disc}(\mathcal{O}) = \operatorname{disc}(\mathcal{O}_K)\mathcal{N}_{\mathcal{O}}^2(\mathfrak{f})$ we get the class number formula in this case.

To use the class number formula we need to compute some approximation of $\operatorname{res} \zeta_{\mathcal{O}}|_{s=1}$ as stated above. The algorithm in [3] derives an approximation to $\operatorname{res} \zeta_{\mathcal{O}_K}|_{s=1}$ from the prime ideals of norm bounded by $C \log(|\operatorname{disc}(\mathcal{O}_K)|)^2$ using a polynomial time algorithm. We note that to compute the primes in $\mathcal{O}_K$ above a rational prime p, we need only the p-maximal over order $\mathcal{O}_p$ of $\mathcal{O}$ which can easily be computed using polynomial time algorithms. As the result of this step we have an approximation to $\operatorname{res} \zeta_{\mathcal{O}_K}|_{s=1}$, as well as approximations to $\phi_{\mathcal{O}_K}(1)$ and $\phi_{\mathcal{O}}(1)$. Bach's algorithm guarantees the precision of the residue, and for the approximations to $\phi_{\mathcal{O}}(1)$ and $\phi_{\mathcal{O}_K}(1)$ we note that the error comes from the maximal ideals of norm greater than $C \log(|\operatorname{disc}(\mathcal{O})|)^2$ above the conductor where C is the constant used in the approximation of the Euler product. Hence the error is close to 1.

To use the data, that is to perform this test, we need (an approximation to) $R(\mathcal{O})$, thus we need to compute fixed point approximations of the $\log |\alpha_i|_j$ and first compute the approximations of the $\log |\beta_i|_j$ where $\beta_i = \prod_j \alpha_i^{V_{i,j}}$. Let R' be the determinant of the matrix

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

whose row vectors are the $(\log|\beta_i|_1, \dots, \log|\beta_i|_r)$. If R' is computed with enough precision, then certifying the result boils down to verifying that $h^* \leqslant h'R' < 2h^*$. Assume that we compute $\widehat{\log|\alpha_i|_j}$ such that $|\widehat{\log|\alpha_i|_j} - \log|\alpha_i|_j| \leqslant 2^{-q}$. Then as each addition induces a loss of one bit of precision, whereas each multiplication by $x \in \mathbb{Z}$ induces a loss of $\log|x|$ bits of precision, we get $\widehat{\log|\beta_i|_j}$ such that $|\widehat{\log|\beta_i|_j} - \log|\beta_i|_j| \leqslant 2^{-q'}$ where $q' = q - (N' + \log|V|)$. To handle the loss of precision due to the determinant computation, we use the following lemma.

LEMMA 4.4. *Let $\Omega = (\omega_1, \dots, \omega_r)$ and $\widehat{\Omega} = (\hat{\omega_1}, \dots, \hat{\omega_r})$ be $r \times r$ matrices with $|\Omega - \hat{\Omega}| \leqslant 2^{-q'}$, then $|\det(\Omega) - \det(\widehat{\Omega})| \leqslant r^{r/2+1}(|\Omega|^{r-1} + 1)2^{-q'}$.*

Proof. We have by multilinearity of the determinant and by Hadamard's inequality

$$\begin{aligned}|\det\hat{\Omega} - \det\Omega| &= \left|\sum_{i=1}^{k} \det(\omega_1, \dots, \omega_{i-1}, \hat{\omega}_i - \omega_i, \hat{\omega}_{i+1}, \dots, \hat{\omega}_k)\right| \\ &\leqslant r^{r/2+1}(|\Omega|^{r-1} + 1)2^{-q'}.\end{aligned}$$

□

As $\log|\alpha_i|_j \leqslant S$, we have $\log|\beta_i|_j \leqslant N'|V|S$. Therefore, by Lemma 4.4, we obtain R' with precision q'' where $q'' = q' - (((r/2) + 1) + \log((N'|V|S)^{r-1} + 1))$.

PROPOSITION 4.5. *Let S be such that $\log|\alpha_i|_j \leqslant S$, vectors $(V_i)_{i \leqslant N'}$ such that the $\beta_i = \prod_j \alpha_j^{V_{i,j}}$ generate the units derived from the kernel vectors of the relation matrix, and a, c such that $N', \log|V| \leqslant L_\Delta(a, c)$. By taking fixed point approximations at a precision q satisfying*

$$q = N' + \log|V| + \left(\frac{r}{2} + 1\right) + \log((N'|V|S)^{r-1} + 1) \in O(L_\Delta(a, c + o(1)) + r\log(S)),$$

we can certify the result in time $O(L_\Delta(a, c + o(1)) + r\log(S))$.

5. *Compact representation*

Assume that we have units $\beta_i = \prod_j \alpha_j^{V_{i,j}}$ where $\log|\alpha_i|_k \leqslant S$, the number of terms N' and $|V|$ satisfy $N', \log|V| \leqslant L_\Delta(a, b)$. We want to rewrite these products as $\beta_i = \prod_j \gamma_j^{l^j}$ where the γ_j have polynomial size, and where the number of terms is polynomial. Such a decomposition is called a compact representation. An exponential algorithm was presented in [35]. In this section, we present a subexponential time algorithm. Our method is summarized in Algorithm 7. In the following, we denote by $\mathrm{Poly}(\Delta)$ the set of values that can be bounded by a polynomial in Δ of degree in $O(n)$. These values have their logarithm bounded by $\log(|\Delta|)^d$ for some $d > 0$.

ALGORITHM 7. Compact representation.

Input: $\beta_j = \prod_{i \leqslant N'} \alpha_i^{V_{i,j}}$ and $(\widehat{\log|\beta_j|_i})_{i \leqslant r+1}$ such that $|\widehat{\log|\beta_j|_i} - \log|\beta_i|_i| \leqslant 2^{-q}$ for $(\beta_j)_{j \leqslant r}$ units with $\log|\beta_j|_i, \log(q), \log|V|, N' \in L_\Delta(a, c + o(1))$ with $0 < a < 1$, $c > 0$, and $l > 0$.

Output: $(\gamma_{i,j}/d_{i,j})$ with $\gamma_{i,j} \in \mathcal{O}$, $d_{i,j} \in \mathbb{Z}_{>0}$, $\|\gamma_{i,j}\|, d_{i,j} \in \mathrm{Poly}(|\Delta|)$ and $\beta_j = \prod_{i \leqslant k} \gamma_{i,j}^{l^i}$, $k \in O(\log|\Delta|)$.

1: $(\widehat{\log|\beta_i|}_1 \dots, \widehat{\log|\beta_i|}_{r+1})_{i \leqslant r} \leftarrow$ LLL reduction of $(\widehat{\log|\beta_j|}_1 \dots, \widehat{\log|\beta_j|}_{r+1})_{j \leqslant r}$.
2: **for** $j = 1 \dots r$ **do**
3: $\quad \beta \leftarrow \beta_j$, $I \leftarrow \mathcal{O}$, $c \leftarrow 1$.
4: $\quad$ Let k_j minimal such that $(1/l^k)\log|\beta_j|_i \leqslant \log\Delta$, $v_i \leftarrow \exp(l^{-k}\log|\beta_j|_i)$.

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

5: **for** $i \leqslant k_j$ **do**
6: $B \leftarrow I^l$, $(w_j)_j \leftarrow (v_j^l)_j$, $c \leftarrow c^l$.
7: $w \leftarrow \sqrt[n]{\prod w_j}$ and $d_i \in \mathbb{Z}_{>0}$ such that $B^{-1} = (1/d_i)C$ for $C \subseteq \mathcal{O}$.
8: Let δ be a 1st LLL-basis element of C with respect to $T_{2,(w_j/w)_j}(\delta) := \sum |\delta|_i^2 w_j^2/w^2$.
9: $I \leftarrow B\delta/d_i$, $(v_j)_{j\leqslant r+1} \leftarrow (w_j \cdot |\delta|_j/d_i)_{j\leqslant r+1}$.
10: $c \leftarrow c\delta$, $\delta_i \leftarrow \delta/d_i$.
11: **end for**
12: $\delta_0 \leftarrow \beta \prod_{1\leqslant i\leqslant k} \delta_i$ modulo primes $\mathfrak{p}_j$ with $\mathcal{N}(\mathfrak{p}_i) \in O(|\Delta|)$ then reconstruct with CRT.
13: $(\gamma_{i,j})_{1\leqslant i\leqslant k} \leftarrow (\delta_i^{-1})_{1\leqslant i\leqslant k}$, $\gamma_{0,j} \leftarrow \delta_0$.
14: **end for**
15: **return** $(\gamma_{i,j})_{1\leqslant j\leqslant r, i\leqslant k}$, $k = \max_j k_j$.

PROPOSITION 5.1. *Algorithm 7 is correct and runs in time* $O(L_\Delta(a, c+o(1)) + n^5 \log(S))$.

Proof. At Step 1, we use the quasi-linear LLL-reduction algorithm of [27] which runs in time $O(L_\Delta(a, c+o(1)) + n^5 \log(S))$ and ensures that $\log |\beta_i|_j \in O(|\Delta|)$. Furthermore, we still have $|\widehat{\log |\beta|_i} - \log |\beta|_i| \leqslant 2^{-q'}$ for some $\log(q') \in L_\Delta(a, c+o(1))$.

Then the loop between Step 4 and 10 preserves the fact that $I = (c)$ and $\mathcal{N}(I) = \prod_i v_i$. It iteratively constructs $(\delta_i)_{1\leqslant i\leqslant k}$ such that $\|\delta_k^{l^i} \dots \delta_i \cdot (\beta)^{l^{k-i}}\| = \sum_j v_j^2 \in \mathrm{Poly}(\Delta)$ with $\|\delta_i\|$ and their denominators in $\mathrm{Poly}(\Delta)$ as well.

Let us prove the conditions on the sizes of the elements. Since $\prod w_j/w = 1$, we get $\|\delta\| \leqslant 2^{O(n)} |\Delta|^{1/2n} \mathcal{N}(C)^{1/n}$ and therefore $\mathcal{N}(I) \in O(|\Delta|^{O(1)})$. We also have

$$T_2(\delta) = \sum_j |\delta|_j^2 = \sum |\delta|_j^2 \frac{w_j^2}{r^2} \frac{r^2}{w_j^2} \leqslant \underbrace{\left(\sum_j |\delta|_j^4 \frac{w_j^4}{w^4}\right)^{1/2}}_{\leqslant T_{2,(w_j/w)}(\delta)} \left(\sum_j \frac{w^4}{w_j^4}\right)^{1/2}$$

by Cauchy–Schwarz. Since $\sum v_j^2 \leqslant \Delta$, we have $v_j \leqslant \Delta^{1/2}$ and therefore $w_j \leqslant \Delta^{l/2}$. From $\prod w_j = N(B) \geqslant 1$, we see $w_j \geqslant \Delta^{-n+1}$, hence $\sum w^4/w_j^4 \in \mathrm{Poly}(\Delta)$. Finally, for the new v_i we have $\sum v_i^2 = T_{2,(w_j/r)_j}(\alpha) \leqslant 2^n \sqrt[n]{d(C)} = O(|\Delta|^{O(1)})$.

The precision issues are the last aspect we need to check. We know fixed points approximations $(1/l^i)\widehat{\log |\beta|}_j$ of $(1/l^i)\log |\beta|_j$ with $L_\Delta(a, c+o(1))$ bits of precision. It turns out that

$$\begin{aligned} |v_i - |\beta|_i| &= |e^{\frac{1}{l^i}\widehat{\log |\beta|}_j} - e^{\frac{1}{l^i}\log |\beta|_j}| = e^{\frac{1}{l^i}\log |\beta|_j} |1 - e^{\frac{1}{l^i}\widehat{\log |\beta|}_j - \frac{1}{l^i}\log |\beta|_j}| \\ &\leqslant e^{\frac{1}{l^i}\log |\beta|_j} |1 - e^{2^{-q'}}| \sim e^{\frac{1}{l^i}\log |\beta|_j} 2^{-q'}. \end{aligned}$$

Since $\log |\beta|_i \in O(|\Delta|)$ and $\log(q') \in L_\Delta(a, c+o(1))$, our precision remains with $L_\Delta(a, c+o(1))$ even when we calculate the v_i. □

6. *Conclusion*

The different parts of the ideal class group and unit group computation algorithm presented in this document run in subexponential time. The bound S on $\log |\alpha_i|_j$ is polynomial because of the bounds on the norms of the algebraic norms of the α_i. To make our claim, we need to rely on the heuristic that sufficiently many relations will span all possible relations. We therefore make the weakest assumption we can afford, which is that a number N' of relations such that

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

N'/N is subexponential will suffice. By the saturation method presented in [7], it is possible to guess all the missing relations from the divisors of the index of the current lattice of relations and the index of the current lattice of units. However, as shown in § 4.2, this index can have a subexponential number of digits, making it impossible to factor in a reasonable time.

HEURISTIC 3. Assume $N = L_\Delta(a, c_1)$ for some $0 < a < 1$ and $c_1 > 0$. Then it suffices to collect N' relations with the methods of § 3 to generate all possible relations if $N'/N = L_\Delta(b, c_2)$ for $0 < b < a$ and $c_2 > 0$.

THEOREM 6.1 (GRH + Heuristic 1 + Heuristic 3). *We have an algorithm to compute the ideal class group and a compact representation of a fundamental system of units of an order $\mathcal{O}$ of discriminant Δ in a number field of degree n in time $L_\Delta(a, c)$ where:*
- *$a = 2/3 + \varepsilon$ for $\varepsilon > 0$ arbitrarily small in the general case;*
- *$a = 1/2$ when $n \leqslant \log(|\Delta|)^{3/4-\varepsilon}$ for $\varepsilon > 0$ arbitrarily small.*

Under the stronger assumption Heuristic 2, and in restricted classes of input, we have the following result.

THEOREM 6.2 (GRH + Heuristic 2 + Heuristic 3). *When $K = \mathbb{Q}[X]/T[X]$ and $d = \log(\max_i |t_i|)$ with $n = n_0 \log(|\Delta|)^\alpha (1 + o(1))$ and $d = d_0 \log(|\Delta|)^{1-\alpha}(1 + o(1))$ for some $0 < \alpha < 1$ and $n_0, d_0 > 0$, then we have an $L_\Delta(a, c)$ algorithm for class group and unit group computation for some $c > 0$ and a satisfying $1 - a \geqslant \alpha \geqslant 1 - 2a$, $a \geqslant 1/3$ and $a \geqslant \alpha$.*

References

1. L. ADLEMAN and J. DEMARRAIS, 'A subexponential algorithm for discrete logarithms over all finite fields', *Advances in cryptology — CRYPTO '93, Proceedings of the 13th Annual International Cryptology Conference,* Santa Barbara, California, USA, August 22–26, 1993, Lecture Notes in Computer Science 773 (ed. D. Stinson; Springer, 1993) 147–158.
2. E. BACH, 'Explicit bounds for primality testing and related problems', *Math. Comp.* 55 (1990) no. 191, 355–380.
3. E. BACH, 'Improved approximations for Euler products', *Number theory,* CMS Conference Proceedings 15 (American Mathematical Society, Providence, RI, 1995) 13–28.
4. J.-F. BIASSE, 'An $L(1/3)$ algorithm for ideal class group and regulator computation in certain number fields', *Math. Comp.* 83 (2014) 2005–2031.
5. J.-F. BIASSE, 'Subexponential time relations in large degree number fields', *Adv. Math. Commun.* Preprint.
6. J.-F. BIASSE, 'Subexponential algorithms for number fields', PhD Thesis, École Polytechnique, Paris, 2011.
7. J.-F. BIASSE and C. FIEKER, 'New techniques for computing the ideal class group and a system of fundamental units in number fields', *Comp. Res. Repository*, Preprint, 2012, arXiv:1204.1294.
8. G. BISSON, 'Endomorphism rings in cryptography', PhD Thesis, LORIA, Nancy, France, 2011.
9. J. BUCHMANN, 'A subexponential algorithm for the determination of class groups and regulators of algebraic number fields', *Séminaire de Théorie des Nombres,* Paris 1988–1989, Progress in Mathematics (ed. C. Goldstein; Birkhäuser, Boston, 1990) 27–41.
10. J. BUCHMANN, C. THIEL and H.C. WILLIAMS, 'Short representation of quadratic integers', *Computational algebra and number theory*, Mathematics and its Applications 325 (Springer, 1995) 159–185.
11. J. CASSELS, *An introduction to the geometry of numbers*, Classics in Mathematics (Springer, 1997); corrected reprint of the 1971 edition.
12. H. COHEN, F. DIAZ Y DIAZ and M. OLIVIER, 'Subexponential algorithms for class group and unit computations', *J. Symbolic Comput.* 24 (1997) no. 3–4, 433–441.
13. A. ENGE, P. GAUDRY and E. THOMÉ, 'An $L(1/3)$ discrete logarithm algorithm for low degree curves', *J. Cryptology* 24 (2011) no. 1, 24–41.
14. P. ERDŐS and P. TURÁN, 'On some problems of a statistical group-theory. I.', *Z. Wahr. Verw. Geb.* 4 (1965) 175–186.
15. N. GAMA and P. NGUYEN, 'Finding short lattice vectors within Mordell's inequality', *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, Victoria, British Columbia, Canada, May 17–20, 2008 (ed. C. Dwork; ACM, 2008) 207–216.

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press

16. D. Gordon, 'Discrete logarithms in GF(p) using the number field sieve', *SIAM J. Discrete Math.* 6 (1993) 124–138.
17. J.L. Hafner and K.S. McCurley, 'A rigorous subexponential algorithm for computation of class groups', *J. Amer. Math. Soc.* 2 (1989) 839–850.
18. G. Hanrot, X. Pujol and D. Stehlé, 'Analyzing blockwise lattice algorithms using dynamical systems', *Advances in Cryptology — CRYPTO 2011 — 31st Annual Cryptology Conference,* Santa Barbara, CA, USA, August 14–18, 2011, Lecture Notes in Computer Science (ed. P. Rogaway; Springer, 2011) 447–464.
19. G. Hanrot and D. Stehlé, 'Improved analysis of Kannan's shortest lattice vector algorithm', *Advances in Cryptology — CRYPTO 2007*, Lecture Notes in Computer Science 4622 (ed. A. Menezes; Springer, 2007) 170–186.
20. D. Jao and V. Soukharev, 'A subexponential algorithm for evaluating large degree isogenies', *Algorithmic number theory*, Lecture Notes in Computer Science 6197 (eds G. Hanrot, F. Morain and E. Thomé; Springer, 2010) 219–233.
21. W.E. Jenner, 'On zeta functions of number fields', *Duke Math. J.* 36 (1969) 669–671.
22. A. Joux, R. Lercier, N. P. Smart and F. Vercauteren, 'The number field sieve in the medium prime case', *Advances in Cryptology — CRYPTO 2006, 26th Annual International Cryptology Conference*, Santa Barbara, California, USA, August 20–24, 2006, Lecture Notes in Computer Science 4117 (ed. Cynthia Dwork; Springer, 2006) 326–344.
23. A. K. Lenstra, H. W. Lenstra Jr., M. S. Manasse and J. M. Pollard, 'The number field sieve', *STOC '90: Proceedings of The Twenty-Second Annual ACM Symposium on Theory of Computing* (ACM, New York, NY, 1990) 564–572.
24. A.K. Lenstra, 'On the calculation of regulators and class numbers of quadratic fields', *Journées Arithmétiques* (Cambridge University Press, 1982) 123–150.
25. D. Micciancio and P. Voulgaris, 'A deterministic single exponential time algorithm for most lattice problems based on Voronoi cell computations', *SIAM J. Comput.* 42 (2013) no. 3, 1364–1391.
26. J. Neukirch, *Algebraic number theory*, Comprehensive Studies in Mathematics (Springer, 1999).
27. A. Novocin, D. Stehlé and G. Villard, 'An LLL-reduction algorithm with quasi-linear time complexity: Extended abstract', *Proceedings of the Forty-third Annual ACM Symposium on Theory of Computing, STOC '11* (ACM, New York, NY, 2011) 403–412.
28. C. P. Schnorr and M. Euchner, 'Lattice basis reduction: improved practical algorithms and solving subset sum problems', *Math. Program.* 66 (1994) no. 2, 181–199.
29. C.P. Schnorr, 'A hierarchy of polynomial time lattice basis reduction algorithms', *Theor. Comput. Sci.* 53 (1987) no. 2–3, 201–224.
30. E. Scourfield, 'On ideals free of large prime factors', *J. Théor. Nombres Bordeaux* 16 (2004) no. 3, 733–772.
31. D. Shanks, 'Class number, a theory of factorization, and genera', *Proceedings of Symposia in Pure Mathematics* 20 (eds W. J. LeVeque and E. G. Straus; American Mathematical Society, 1969) 415–440.
32. D. Shanks, 'The infrastructure of a real quadratic field and its applications', *Proceedings of the 1972 Number Theory Conference* (American Mathematical Society, 1972) 217–224.
33. N. Smart and F. Vercauteren, 'Fully homomorphic encryption with relatively small key and ciphertext sizes', *Public Key Cryptography — PKC 2010*, Lecture Notes in Computer Science 6056 (eds P. Nguyen and D. Pointcheval; Springer, 2010) 420–443.
34. A. Storjohann, 'Algorithms for matrix canonical forms', PhD Thesis, Department of Computer Science, Swiss Federal Institute of Technology – ETH, 2000.
35. C. Thiel, On the complexity of some problems in algorithmic algebraic number theory, PhD Thesis, Universität des Saarlandes, 1995.

Jean-François Biasse
Computer Science Department and Mathematics Department
University of Calgary
2500 University Drive NW
Calgary, Alberta T2N 1N4
Canada
biasse@lix.polytechnique.fr

Claus Fieker
University of Kaiserslautern
Fachbereich Mathematik
Postfach 3049
67653 Kaiserslautern
Germany
fieker@mathematik.uni-kl.de

https://doi.org/10.1112/S1461157014000345 Published online by Cambridge University Press