Manuscript submitted to
AIMS' Journals
Volume **X**, Number **0X**, XX **200X**

doi:10.3934/xx.xxxxxxx

pp. **X–XX**

AN ALGORITHM FOR SOLVING THE PRINCIPAL IDEAL PROBLEM WITH SUBFIELDS

JEAN-FRANÇOIS BIASSE, CLAUS FIEKER, TOMMY HOFMANN, AND WILLIAM YOUMANS

Jean-François Biasse*[1], Claus Fieker[2], Tommy Hofmann[3], and William Youmans[1]

1 University of South Florida
Center for Cryptographic Research
4202 E Fowler Ave, Tampa, FL 33620, USA

2 University of Kaiserslautern
Department of Mathematics
Postfach 3049, Kaiserslautern,67653,Germany

3 University of Siegen
Department of Mathematics
Walter-Flex-Straße, Siegen, 57072, Germany

(Communicated by the associate editor name)

Abstract. The principal ideal principal (PIP) is the problem of deciding whether a given ideal of a number field is principal and, if it is, of finding a generator. Solving the PIP applies to solving major computational tasks in number theory. It is also connected to the search of approximate short vectors in the so-called *ideal lattices* which is a crucial problem in cryptography. In this paper, we present a novel application of norm relations to utilize information from subfields to solve the PIP in fields of degree larger than 1800.

1. Introduction.

Background. Given an ideal of a number field, the principal ideal problem (PIP) is the problem of deciding whether it is principal and if so, of finding one of its generators. This is a computational problem of high importance in computational number theory. It is a subroutine (and often the bottleneck) of algorithms for computing invariants of number fields such as relative class groups, S-units and S-class groups or ray class groups (see for example the work of Simon [38]). Additionally, the PIP has recently received considerable attention from the cryptography community for its connections to the security of cryptosystems based on the so-called *ideal lattices*. Indeed, a series of schemes based their security on the hardness of finding a short generator of a principal ideal in a number field (usually a cyclotomic field). This was the case in particular for the homomorphic encryption scheme of

2020 *Mathematics Subject Classification.* Primary: 11Y16, 20C05, 11R32; Secondary: 11R29, 11R04, 11Y40, 11R18, 11R27.

Key words and phrases. Ideal class groups, Brauer relations, Principal ideal problem, approximate short vectors.

The first author is supported by NSF grant 1846166.

* Corresponding author: Jean-François Biasse.

Vercauteren and Smart [39], the multilinear maps of Garg, Gentry and Halevi [26], and a proposition for post-quantum secure encryption by the British GCHQ [17]. Finding a (not necessarily small) generator of a principal ideal is the computational bottleneck of these attacks. In cyclotomic fields, once a generator is found, a smaller one can be derived by using lattice techniques due to Cramer, Ducas, Peikert and Regev [19], thus providing a solution to the Short-Principal Ideal Problem (SPIP). It was established later that solving the PIP was also connected to the search for approximate short vectors in arbitrary ideal lattices (i.e. not necessarily those arising from principal ideals) in cyclotomic fields. In particular, Cramer, Ducas and Wesolowski [20] described a reduction from the search of solutions to the γ-Shortest Vector Problem (γ-SVP) where $\gamma \in e^{\tilde{O}(\sqrt{n})}$ for n the degree of the field to the resolution of the PIP. The γ-SVP is an approximation of the Shortest Vector Problem (SVP) consisting in searching for vectors in an input lattice $\mathcal{L}$ with length within a factor γ of the shortest non-zero vector of $\mathcal{L}$. Such approximate short vectors cannot be found with asymptotically efficient lattice reduction methods such as LLL [30], which means that the resolution of the PIP is connected to the search for non-trivial approximations of short vectors in lattices, a long standing fundamental problem in algorithmic theory with essential applications to cryptography. In particular, a solution to SPIP in a principal ideal $\mathfrak{a}$ is a solution to γ-SVP where $\gamma \in e^{\tilde{O}(\sqrt{n})}$. In general lattices, the most efficient method for solving this problem is the BKZ algorithm [37] whose time complexity is in $e^{\tilde{O}(\sqrt{n})}$. Cryptosystems based on ideal lattices rely on the assumption that γ-SVP in ideals of cyclotomic fields is not significantly easier than in general lattices (in particular for polynomial γ). The results in this paper illustrate that even without quantum computers, there are families of cyclotomic fields in which certain instances of γ-SVP (for subexponential γ) can be solved asymptotically faster than BKZ. This shows that certain instances of γ-SVP in ideals of cyclotomic fields are not as difficult as in general lattices.

Prior Work. The known subexponential methods for solving the PIP in ideals of a number field K rely on the computation of the class group of K. The subexponential strategy for the computation of the class group of an imaginary quadratic field was described in 1989 by Hafner and McCurley [28]. The expected running time of this method is

$$L_\Delta(1/2, \sqrt{2} + o(1)) = e^{\left(\sqrt{2}+o(1)\right)\sqrt{|\Delta_K| \log\log|\Delta_K|}},$$

where Δ_K is the discriminant of the field. Buchmann [16] generalized this result to the case of infinite classes of number fields with fixed degree. Practical improvements to Buchmann's algorithm were presented in [18] by Cohen, Diaz Y Diaz and Olivier. In [5, 10], Biasse and Fieker showed that there was a heuristic subexponential algorithm for the computation of the ideal class group in all classes of number fields, and that it could be used to solve the PIP. The methods of [10] can be specialized to the case of cyclotomic fields for a better asymptotic complexity [7] (heuristically in $e^{\tilde{O}(\sqrt{\log|\Delta_K|})}$). This complexity can be brought even further down (as low as $e^{\tilde{O}(\sqrt[3]{\log|\Delta_K|})}$ assuming a one-time subexponential precomputation on the field [6]).

A turning point in the development of algorithms for solving the PIP was achieved when Bauch, Bernstein, de Valence, Lange and van Vredendaal [3] showed how to recursively solve the PIP in fields of the form $K = \mathbb{Q}(\sqrt{d_1}, \ldots, \sqrt{d_n})$ (the multiquadratic fields). Depending on the family of d_i chosen, this method can have asymptotic complexity as low as polynomial in the logarithm of the discriminant of the field. This method was successfully adapted to calculation of S-unit groups and ideal class groups by Biasse and van Vredendaal [15] who proved that it had asymptotic run time in $\mathrm{Poly}(\log(|\Delta_K|))e^{\tilde{O}(\sqrt{\log|d|})}$ with $d = d_1 \cdots d_n$, under the generalized Riemann hypothesis (GRH) and an assumption on the distribution of certain families of characters. The main idea allowing a recursive computation in the subfields of relative degree 2 was to find a norm relation implying that the square of any element in K was the product of elements coming from 3 subfields. In another direction, the method of [3] was adjusted by Lesavourey, Plantard and Susilo [32] to the case of multicubic fields. Recent work from Biasse, Fieker, Hofmann and Page [13] generalized this concept of norm relation involving subfields to a large variety of number fields (essentially consisting of fields whose Galois group is "far from" being cyclic). Among other computational tasks, they showed how to leverage these relations to compute S-unit groups and ideal class groups recursively using subfields.

Our contribution. We use the norm relations construction technique introduced by Biasse, Fieker, Hofmann and Page [13] to efficiently solve the PIP recursively in non-cyclic number fields. This framework includes the prior work of [3] on multiquadratics and extends it to a significantly larger variety of fields. The prior work of [13] allows the recursive computation of S-unit groups from subfields, which in turn can be used to solve the PIP. In this paper, we use norm relations to solve the PIP without having to compute S-unit groups. This results in a significant practical speed-up over the direct application of [13]. In addition, we are able to solve the PIP in fields of degree significantly larger than the previous state of the art. More specifically, the main technical contributions of this paper are:

- An algorithm using the norm relations given in [13] to solve directly the PIP using subfields (Section 4). It avoids calculating S-units, and it uses an efficient algorithm for computing the roots modulo unit groups of high degree fields arising from norm relations (Section 5).
- An implementation of our method that uses the computer algebra package Hecke [23] based on the programming language Julia which successfully solved the PIP in cyclotomic fields of degree 400, 864, and 1800 respectively (Sections 7). We also report the computation of short generators of ideals (SPIP) in cyclotomic fields of degree 400 and 864.
- A complexity analysis of the proposed method, together with the description of an infinite family of cyclotomic fields in which methods based on norm relations solve the PIP with strong subexponential complexity $\mathrm{SUBEXP} = \cap_{\varepsilon>0} \mathrm{DTIME}(2^{n^{\varepsilon}})$. (Section 6).

The numerical results presented in this paper are significantly better than the previous state of the art. Indeed, as the degree grows, class group and PIP techniques previously available become rapidly impractical. Even the previous ad-hoc implementations for setting prior records are limited to degrees significantly less to what we achieved in this paper. For comparison, the subexponential method described in [7] only contained implementations of certain subroutines of the resolution of the PIP in $\mathbb{Q}(\zeta_{512})$ (of degree 256): namely the computation of the class group of $\mathbb{Q}(\zeta_{512})^+$ (of degree 128), and the Gentry–Szydlo subroutine [27]. The highest degree achieved by the recursive PIP implementation in [3] in multiquadratic fields was 256, and this implementation was restricted to only multiquadratic fields. Our implementation applies to a much wider class of number fields, and has been tested and applied in significantly higher degrees.

In [13], norm relations were used to compute the class group of $\mathbb{Q}(\zeta_{6552})$ of degree 1728 using a PARI/GP implementation in just over 4 hours on a single core. While this does illustrate the potential of norm relations, it does unfortunately not apply to S-unit computation or PIP resolution. Indeed, the successful class group computation in $\mathbb{Q}(\zeta_{6552})$ was achieved without having to return outputs that are field elements. When computing S-units, or solving the PIP, expensive root calculations have to occur in the number field, which is the bottleneck of the overall calculation. The Julia-based implementation of the methods presented in this paper (which is separate from the PARI/GP code from [13]) contains efficient methods for root computation that make the resolution of the PIP possible in large dimension.

2. High level overview.

Let K be a number field with ring of integers $\mathcal{O}_K$, and $\mathfrak{a}$ be an input principal ideal. Suppose that there are subfields $(K_i)_{i\leq l}$ of K such that

$$\forall x \in K^\times,\ x^d = \prod_{i=1}^{l} \mathrm{N}_{K/K_i}(x^{b_i})^{a_i} \tag{1}$$

for some integers d, a_i, b_i, $1 \leq i \leq l$. The methods proposed in this paper efficiently reduce the resolution of the PIP with input $\mathfrak{a}$ to instances of the PIP in the subfields $(K_i)_{1\leq i\leq l}$. Identities of the form (1) are called *norm relations*. Criteria to decide whether non-trivial norm relations exist were presented in [13], as well as efficient methods to compute optimal norm relations (with respect to the degree of the subfields K_i).

In [13], algorithms for using norm relations to compute the ideal class group $\mathrm{Cl}(\mathcal{O}_K)$, as well as S-unit groups [13, Alg. 4.16] were presented. A PARI/GP implementation of the class group method was provided. It only works on a subset of instances where computations in K can be totally avoided. In particular, this rules out the computation of S-unit groups which are subgroups of $K^\times$. From a theoretical standpoint though, the computation of S-unit groups does allow the resolution of the PIP as follows: Given the input ideal $\mathfrak{a}$, we begin by enumerating $\alpha \in \mathfrak{a}$ that are small combinations of an LLL-reduced basis of $\mathfrak{a}$ until $(\alpha)/\mathfrak{a} = \mathfrak{p}$ a prime ideal. Then let $S = \{\mathfrak{p}^\sigma \mid \sigma \in \mathrm{Gal}(K/\mathbb{Q})\}$ be the set of all conjugates of $\mathfrak{p}$ under the action of the Galois group $\mathrm{Gal}(K/\mathbb{Q})$ of K. Using [13, Alg. 4.16], we then compute a generating set $\alpha_1, \ldots \alpha_{r+s}$ of the S-unit group where r is the rank of the unit group $\mathcal{O}_K^\times$ and $s = |S|$, together with vectors $\vec{v}_1, \ldots, \vec{v}_{r+s} \in \mathbb{Z}^s$ describing the valuations of the α_i a the primes in S. Finally, we solve a linear system to find $\vec{x} \in \mathbb{Z}^{r+s}$ such that $\sum_i x_i\vec{v}_i$ is the vector with zeros everywhere except for a 1 in the entry corresponding to $\mathfrak{p}$. Then $\prod_i \alpha_i^{x_i}$ is a generator of $\mathfrak{p}$, and $g = \alpha \cdot \prod_i \alpha_i^{-x_i}$ is a generator of $\mathfrak{a}$, which solves the PIP.

The methods from [13, Alg. 4.16] to compute S-unit groups require the computation of a significant number of d-th roots in K, where d is defined in (1). While this only incurs extra polynomial factors to the asymptotic complexity, it effects the performance of the practical implementations, and prevents them from reaching the record-breaking input sizes presented in this paper.

We present a new method to leverage norm relations to solve the PIP without having to use the reduction from S-unit group computations mentioned above. In essence, our methods generalize the multiquadratic approach of [3] which relied on the fact that the square of each element in a multiquadratic field could be expressed as the product of elements from 3 subfields. We briefly recall the main results on norm relations in Section 3. A relation of the form (1) implies that

$$\mathfrak{a}^d = \prod_{i=1}^{l} \mathrm{N}_{K/K_i}(\mathfrak{a}^{b_i})^{a_i}\mathcal{O}_K.$$

If the ideal $\mathfrak{a}$ is principal, then so are the subfield ideals $\mathrm{N}_{K/K_i}(\mathfrak{a}^{b_i})$. Solving the corresponding PIP in the subfields K_i gives us generators α_i of the ideals $\mathrm{N}_{K/K_i}(\mathfrak{a}^{b_i})$. Then $\beta = \prod_i \alpha_i^{a_i} \in K$ generates the ideal $\mathfrak{a}^d$. Unfortunately, this does not readily give us a generator of $\mathfrak{a}$. In Sections 4 and 5, we show how to use a saturation-like method to find a suitable unit $u \in K^\times$ such that $\beta \cdot u$ is a d-th power. Then, we compute the d-th root of $\beta \cdot u$ which yields a generator of $\mathfrak{a}$. This latter step can be computationally expensive, but only needs to be performed once. We show how to solve it by writing subfield elements efficiently in compact representation (Section 5.3). We describe another crucial practical improvement in Section 5: We show that it is not necessary to compute the full unit group of K, therefore avoiding many costly saturation steps.

We analyze the complexity of our algorithms, and we describe an infinite family of cyclotomic fields in which we solve PIP in time $2^{n^{o(1)}}$ assuming only GRH (Section 6). We provide numerical results in Section 7, which include the resolution of the SPIP in degree 864 and of the PIP in degree 1800. Finally, in Section 8 we compare our methods with the direct use of the S-unit algorithms of [13] to solve the PIP sketched above. The supplementary material presents background information on prior art. The source code of our implementation is also supplied with this submission.

3. Norm relations.

In this section, we recall some facts about norm relations and their existence and refer the reader to [13] for details. Let K be a Galois algebraic number field with Galois group $G = \mathrm{Gal}(K/\mathbb{Q})$. For a subgroup $H \leq G$ we denote by $N_H = \sum_{h\in H} h \in \mathbb{Q}[G]$ the norm of H as an element of the group algebra $\mathbb{Q}[G]$. A *norm relation of* G is an equality of the form

$$1 = \sum_{i=1}^{l} a_i N_{H_i} b_i \tag{2}$$

in $\mathbb{Q}[G]$ with $a_i, b_i \in \mathbb{Q}[G]$ and $1 \neq H_i \leq G$ subgroups. By clearing denominators, a norm relation can always be written as

$$d = \sum_{i=1}^{l} a_i N_{H_i} b_i \tag{3}$$

with $d \in \mathbb{N}_{>0}$ minimal such that $a_i, b_i \in \mathbb{Z}[G]$. We call d the *denominator* of the norm relation.

The existence of such a norm relation for a number field implies relations between arithmetic objects of the field K and its subfields (see [13]). In the present paper, we will use the fact that Equation (3) implies that for all $x \in K^\times$ we have

$$x^d = \prod_{i=1}^{l} \mathrm{N}_{K/K^{H_i}}(x^{b_i})^{a_i}, \tag{4}$$

where $K^H = \{x \in K \mid \sigma(x) = x \text{ for all } \sigma \in H\}$ is the fixed field of H, and

$$x^a = \prod_{g\in G} g(x)^{a_g} \text{ for all } x \in K^\times \text{ and } a = \sum_{g\in G} a_g g \in \mathbb{Z}[G].$$

We will most often use an equality of the form (4) when referring to a norm relation. Let now $\mathfrak{a}$ be a fractional ideal of K. From [33, Chapter III, §1, Proposition 1.6] it follows that for a subgroup $H \leq G$ the following relation holds: $\mathrm{N}_{K/K^H}(\mathfrak{a})\mathcal{O}_K = \prod_{\sigma\in H} \sigma(\mathfrak{a}) = \mathfrak{a}^{N_H}$. In particular, from Equation (3), we also obtain

$$\mathfrak{a}^d = \prod_{i=1}^{l} \mathrm{N}_{K/K^H}(\mathfrak{a}^{b_i})^{a_i}\mathcal{O}_K. \tag{5}$$

Example 1. Let $G = C_2 \times C_2 = \langle \sigma, \tau \rangle$. Then we have the norm relation $2 = N_{\langle\sigma\rangle} + N_{\langle\tau\rangle} - \sigma N_{\langle\sigma\tau\rangle}$. This is the norm relation used implicitly in both [3] and [15].

Due to Funakura [25] we have the following simple criterion for the existence of norm relations.

Theorem 3.1 ([13, Theorem 2.27])**.** *Let G be a finite abelian group, and write $G \cong C \times Q$ where C is the largest cyclic factor of G.*

1. *The group G admits a norm relation with denominator 1 if and only if $|Q|$ is divisible by at least two distinct primes. If the condition is satisfied, then G admits a norm relation with $a_i \in \mathbb{Z}$, denominator 1, and where all H_i satisfy that G/H_i is a p_i-group times a cyclic group, for some prime number p_i.*
2. *Assume that Q is a p-group. Then G admits a norm relation if and only if $Q \neq 1$. If the condition is satisfied, then G admits a norm relation with $a_i \in \mathbb{Z}$, denominator a power of p and where all H_i satisfy that G/H_i is a cyclic group.*

We will exploit Theorem 3.1 in Section 7 to construct large degree number fields for which it is possible to solve the principal ideal problem efficiently.

4. **Principal ideals and norm relations.** We now explain our new strategy to solve the PIP using norm relations without S-unit computations. Let K be a Galois number field and $G = \mathrm{Gal}(K/\mathbb{Q})$. Throughout this section we assume that G admits a norm relation involving the subgroups $\{H_1, \dots, H_l\}$. Thus there exist $a_i, b_i \in \mathbb{Z}[G]$, $1 \leq i \leq l$, and $d \in \mathbb{Z}$ with $d = \sum_{i=1}^{l} a_i H_i b_i$ Recall that this implies that for an element $x \in K^\times$ and a fractional ideal $\mathfrak{a}$ of K the following also holds:

$$x^d = \prod_{i=1^l} \mathrm{N}_{K/K_i}(x^{b_i})^{a_i} \quad \text{and} \quad \mathfrak{a}^d = \prod_{i=1}^{l} \mathrm{N}_{K/K_i}(\mathfrak{a}^{b_i})^{a_i}\mathcal{O}_K. \tag{6}$$

where K_i denotes the fixed field of H_i.

Lemma 4.1. *Let $\mathfrak{a}$ be a fractional ideal of K. If $\mathfrak{a}$ is principal, then $\mathrm{N}_{K/K_i}(\mathfrak{a}^{b_i})$ is principal for all $1 \leq i \leq l$. If $d = 1$, then the converse also holds and a generator of $\mathfrak{a}$ is given by $\prod_{i=1}^{l} \alpha_i^{a_i}$, where $\alpha_i\mathcal{O}_K = \mathrm{N}_{K/K_i}(\mathfrak{a}^{b_i})$, $1 \leq i \leq l$.*

The previous lemma shows that if the denominator d is equal to 1, then solving the principal ideal problem in K is equivalent to solving the principal ideal problems in the subfields K_i, $1 \leq i \leq l$. In case the denominator is not equal to 1, the situation is more complicated. Indeed, we can only find α such that $\mathfrak{a}^d = \alpha\mathcal{O}_K$. If α is a d-th power, say $\beta^d = \alpha$, then β generates $\mathfrak{a}$. Otherwise, if α is not a d-th power, we need to find another generator α' of $\mathfrak{a}^d$ that is a d-th power. This is done by multiplying α by a well-chosen unit.

Definition 4.2. Let $U \subseteq K^\times$ be a multiplicative group and $\beta \in K^\times$. We say that β is a *d-th power modulo U* or that *β has a d-th root modulo U*, if there exists $u \in U$ such that $u\beta$ is a d-th power, i.e., $u\beta \in (K^\times)^d$.

From the preceding discussion it follows that if $\mathfrak{a}^d = \beta\mathcal{O}_K$, then $\mathfrak{a}$ is principal if and only if β is a root modulo $U = \mathcal{O}_K^\times$. Moreover if $u\beta = \alpha^d$ for some $u \in \mathcal{O}_K^\times$, then $\mathfrak{a} = \alpha\mathcal{O}_K$. Working with the full unit group $U = \mathcal{O}_K^\times$ can be expensive in practice. In the following we improve upon this by showing that in our situation we can often pick a smaller group U generated by subgroups of the unit groups $\mathcal{O}_{K_i}^\times$. We begin by showing that one can restrict to full rank subgroups with index coprime to d.

Lemma 4.3. *Assume that $U \subseteq K^\times$ is a multiplicative group, $\beta \in K^\times$ and $d \in \mathbb{Z}$. Further let $V \subseteq U$ be a subgroup of finite index with $[U : V]$ coprime to d. Then β is a d-th power modulo U if and only if β is a d-th power modulo V.*

Proof. Let $k = [U : V]$ and $a, b \in \mathbb{Z}$ such that $ad + bk = 1$. Assume that there exists $u \in U$ such that $u\beta \in (K^\times)^d$. As $u\beta = (u^a)^d(u^k)^b\beta$ and $v = (u^k)^b \in V$, we have $v\beta = u\beta/(u^a)^d \in (K^\times)^d$, thus showing that β is a d-th power modulo V. The other implication is clear. □

We can now show that in the presence of norm relations, it is sufficient to work with a multiplicative group generated by units from the involved subfields. In fact, not even the full unit groups of the subfields are necessary, but just subgroups with index coprime to d.

Proposition 1. *Let $\mathfrak{a}$ be a fractional ideal satisfying (6). Assume that the ideal $\mathrm{N}_{K/K_i}(\mathfrak{a}^{b_i}) = \alpha_i\mathcal{O}_{K_i}$ is principal for all $1 \leq i \leq l$ and let $\beta = \prod_{i=1}^l \alpha_i^{a_i}$. Consider the multiplicative group $W = (\mathcal{O}_{K_1}^\times)^{a_1} \cdots (\mathcal{O}_{K_l}^\times)^{a_l} \subseteq \mathcal{O}_K^\times$. Let $V \subseteq W$ be a subgroup of finite index with $[W : V]$ coprime to d and $V_i \subseteq \mathcal{O}_{K_i}^\times$ subgroups of finite index with $[\mathcal{O}_{K_i}^\times : V_i]$ coprime to d. Then the following are equivalent:*

(a) The ideal $\mathfrak{a}$ is principal.
(b) The element β is a d-th power modulo $\mathcal{O}_K^\times$.
(c) The element β is a d-th power modulo W.
(d) The element β is a d-th power modulo V.
(e) The element β is a d-th power modulo $V_1^{a_1} \cdots V_l^{a_l}$.

If we have $a_i \in \mathbb{Z}$ for all $1 \leq i \leq l$, then we can use $W = \mathcal{O}_{K_1}^\times \cdots \mathcal{O}_{K_l}^\times$ in (d) and $V_1 \cdots V_l$ in (e).

Proof. (a) ⇔ (b): Clear.

(a) ⇔ (c): Assume $\mathfrak{a} = \alpha\mathcal{O}_K$ is principal. As $\mathrm{N}_{K/K_i}(\alpha^{b_i})\mathcal{O}_{K_i} = \mathrm{N}_{K/K_i}(\mathfrak{a}^{b_i}) = \alpha_i\mathcal{O}_{K_i}$, there exist units $u_i \in \mathcal{O}_{K_i}^\times$ such that $\mathrm{N}_{K/K_i}(\alpha^{b_i}) = u_i\alpha_i$. Thus

$$\underbrace{u_1^{a_i} \cdots u_l^{a_l}}_{\in W} \cdot \beta = \prod_{i=1}^l (u_i\alpha_i)^{a_i} = \prod_{i=1}^l \mathrm{N}_{K/K_i}(\alpha^{b_i})^{a_i} = \alpha^d \in (K^\times)^d$$

and β is a d-th power modulo W. Conversely if β is a d-th power modulo W, it is also a d-th power modulo $\mathcal{O}_K^\times$ and hence $\mathfrak{a}$ is principal.

(c) ⇔ (d): Lemma 4.3.

(d) ⇔ (e): Since the V_i have index coprime to d, it follows that $[W : V_1^{a_1} \cdots V_l^{a_l}]$ is coprime to d. Hence the result follows again by Lemma 4.3. □

Proposition 1 leads to Algorithm 1 to solve the PIP using norm relations. In Section 5, we will show how to perform Steps 9 to 13 efficiently.

```
Input  : A fractional ideal $\mathfrak{a}$ of $K$ satisfying (6)
Output: Whether $\mathfrak{a}$ is a principal ideal and a generator in case it is
1  $y \leftarrow 1$;
2  for $i \leftarrow 1$ to $l$ do
3      if $\mathrm{N}_{K/K_i}(\mathfrak{a}^{b_i})$ is principal then
4          Find a generator $\alpha_i \in K_i$ of $\mathrm{N}_{K/K_i}(\mathfrak{a}^{b_i})$;
5      else
6          return: $\mathfrak{a}$ is not principal.
7      end
8  end
9  $\beta \leftarrow \alpha_1^{a_1} \cdots \alpha_l^{a_l}$ //$\beta$ generates $\mathfrak{a}^d$.;
10 Compute $U = V_1^{a_1} \cdots V_l^{a_l}$, where the $V_i$ are subgroups $\mathcal{O}_{K_i}^\times$ with index
     coprime to $d$. // Details in Section 5;
11 if $\beta$ is a $d$-th power modulo $U$ then
12     return: $\alpha \in K^\times$ such that $\beta/\alpha^d \in U$;
13 else
14     return: $\mathfrak{a}$ is not principal;
15 end
```

Algorithm 1: Strategy for solving the PIP from norm equations

Remark 1. The idea of reducing the principal ideal problems to subfields using relative norms and the existence of d-th powers was already considered in [3] for multiquadratic and in [32] for multicubic fields. While not formulated using the notion of norm relations, in both works criterion 1 (b) is used to decide the principal ideal problem in the field K. In particular, the full unit group had to be computed via saturation.

In contrast to the aforementioned papers, the use of Proposition 1 (c) allows us to avoid the computation of the full unit group $\mathcal{O}_K^\times$ of K (in most cases, see Section 5). Actually Proposition 1 (e) allows us to avoid the computation of the full unit groups in the subfields themselves. All that is required are subgroups whose index is finite and coprime to d. This results in a significant practical speed-up.

5. Determining roots modulo units.

In this section, we describe efficient algorithms to perform Steps 9 to 13 of Algorithm 1. Let K be an algebraic number field and $U \subseteq K^\times$ a finitely generated multiplicative group and $\beta \in K^\times$. Throughout this section we assume that $U \cap \langle \beta \rangle = \{1\}$ and that U is specified by a finite number of generators. In our application this is true, since $U \subseteq \mathcal{O}_K^\times$ and β is a generator of a non-trivial ideal. We want to decide whether β is a d-th power modulo U, that is, decide whether there exists $u \in U$ such that $u \cdot \beta \in (K^\times)^d$. In the following, for a non-zero prime ideal $\mathfrak{p}$ of $\mathcal{O}_K$ we will denote by $K_\mathfrak{p}$ the $\mathfrak{p}$-adic completion of K, by $v_\mathfrak{p}$ the $\mathfrak{p}$-adic valuation and by $k_\mathfrak{p} = \mathcal{O}_K/\mathfrak{p} \cong \mathcal{O}_{K_\mathfrak{p}}/\mathfrak{p}\mathcal{O}_{K_\mathfrak{p}}$ the residue field at $\mathfrak{p}$. We use a bar notation to denote cosets of various multiplicative groups, and $\langle X \rangle$ to denote the subgroup generated by X.

Lemma 5.1. *Assume that $U \subseteq K^\times$ is a multiplicative group and $d = a \cdot b$ with $\gcd(a, b) = 1$. Then β is a d-th power modulo U if and only if β is an a-th and a b-th power modulo U.*

Proof. Since one of the implications is trivial, let us assume that β is an a-th and a b-th power modulo U, say $\beta = u\gamma^a = u_0\gamma_0^b$. Since a and b are coprime there exist $r, s \in \mathbb{Z}$ with $1 = ra + sb$. Thus

$$\beta = \beta^{ra}\beta^{sb} = (u_0\gamma_0^b)^{ra}(u\gamma^a)^{sb} = u_0^{ra}u^{sb}(\gamma_0^r\gamma^s)^d \in U \cdot (K^\times)^d.$$

□

Thus from now on we will assume that d is a prime power. The method we want to describe will employ local computations to detect global powers. This a well known technique in computational algebraic number theory, used for example in the class and unit group computation of number fields ([36, Section 5.7]) or the number field sieve ([1]). Note that, in contrast to previous applications of this technique, in our case the number d is in general not a prime. As a consequence, we will rely on the Grunwald–Wang theorem (see [2, Chapter X] or [34, Chapter IX, §1]) and therefore have to consider the following dichotomy. For $k \in \mathbb{Z}_{\geq 1}$ denote by ζ_k a primitive k-th root of unity and set $\eta_k = \zeta_k + \zeta_k^{-1}$. Let $s \geq 2$ be an integer such that $\eta_s \in K$ but $\eta_{s+1} \notin K$. Moreover let S be a finite set of prime ideals of $\mathcal{O}_K$. Recall that d is a prime power. We say that we are in the *bad case* when the following conditions are simultaneously satisfied

1. The number $d = 2^t$ is even and $t > s$.
2. The elements $-1, 2 + \eta_s$ and $-(2 + \eta_s)$ are non-squares in K.
3. We have
$$\{\mathfrak{p} \mid 2 \in \mathfrak{p} \text{ and } -1, 2 + \eta_s \text{ and } -(2 + \eta_s) \text{ are non-squares in } K_\mathfrak{p}\} \subseteq S.$$

If we are not in the bad case, we say that we are in the *good case*. The terminology is explained by the theorem of Grunwald–Wang, which gives the following connection between global and local d-th powers.

Theorem 5.2 (Grunwald–Wang)**.** *Consider the canonical map*

$$K^\times/(K^\times)^d \longrightarrow \prod_{\mathfrak{p} \notin S} K_\mathfrak{p}^\times/(K_\mathfrak{p}^\times)^d.$$

If we are in the good case, this map is injective. If we are in the bad case, the kernel of the map is $\langle \bar{\eta}_s \rangle \cong \mathbb{Z}/2\mathbb{Z}$.

Remark 2.

1. Given d, it is straightforward to test conditions (1) and (2). To test condition (3), it is sufficient to determine all prime ideals $\mathfrak{p}$ lying over 2 such that $-1, 2 + \eta_s$ and $-(2 + \eta_s)$ are non-squares in $K_\mathfrak{p}$. Being locally a square can be checked using the so-called quadratic defect [35, §63.A], which can be computed using an efficient algorithm due to Kirschmer [29, Algorithm 3.1.3]. Thus given K, d and S, we can always check whether we are in the good case or not.
2. Although the conditions for being in the bad case look rather complicated, this situation is not as rare as it might appear. More precisely, if K is linear disjoint from the cyclotomic field $\mathbb{Q}(\zeta_8)$, then we are always in the bad case for $d = 2^t$, $t \geq 3$. Thus for almost all fields we are in the bad case at the prime 2.

We now explain how to decide whether β is a d-th power modulo U depending on whether or not we are in the good case.

5.1. **The good case.** In this section we will present a method to test if an element is a d-th power modulo a multiplicative group in the good case of Grunwald–Wang. To detect local powers, we will make use of the following statements. Recall that for a set S of prime ideals of $\mathcal{O}_K$ we denote by $\mathcal{O}_{K,S}$ the ring of S-integers, that is, the elements $x \in K$ with $v_\mathfrak{p}(x) \geq 0$ for all $\mathfrak{p} \notin S$, and $\mathcal{O}_{K,S}^\times$ the group of S-units, i.e., the elements $x \in K^\times$ such that $v_\mathfrak{p}(x) = 0$ for all $\mathfrak{p} \notin S$.

Proposition 2 ([13, Proposition 4.5])**.** *Assume that $\mathfrak{p}$ is a non-zero prime ideal with $d \notin \mathfrak{p}$ and let $\varpi \in K$ be a local uniformizer at $\mathfrak{p}$, that is, an element with $v_\mathfrak{p}(\varpi) = 1$. Then the map*

$$K_\mathfrak{p}^\times/(K_\mathfrak{p}^\times)^d \longrightarrow \mathbb{Z}/d\mathbb{Z} \times k_\mathfrak{p}^\times/(k_\mathfrak{p}^\times)^d, \quad \bar{x} \longmapsto (\overline{v}, \overline{x\varpi^{-v}}) \text{ where } v = v_\mathfrak{p}(x),$$

is an isomorphism.

Proposition 3. *Assume that we are in the good case of Grunwald–Wang. For a multiplicative finitely generated subgroup $V \subseteq K^\times$ we have*

$$(V \cap (K^\times)^d)/V^d = \bigcap_{d \notin \mathfrak{p}} \ker(V/V^d \to \mathbb{Z}/d\mathbb{Z} \times k_\mathfrak{p}^\times/(k_\mathfrak{p}^\times)^d).$$

There exists $c_0 \in \mathbb{R}_{>0}$ (depending on K, V and d) such that

$$(V \cap (K^\times)^d)/V^d = \bigcap_{d \notin \mathfrak{p}, \mathrm{N}(\mathfrak{p}) \leq c_0} \ker(V/V^d \to \mathbb{Z}/d\mathbb{Z} \times k_\mathfrak{p}^\times/(k_\mathfrak{p}^\times)^d).$$

Proof. The first part is [13, Proposition 4.6]. As V is finitely generated, V/V^d is a finitely generated $(\mathbb{Z}/d\mathbb{Z})$-module. Thus V/V^d is Artinian and the existence of c_0 follows from the first part. □

Putting everything together, we arrive at the following criterion for detecting whether an element β is a d-power modulo U, and if so, for computing $u \in U$ such that $\beta \cdot u$ is a d-th power. Note that Proposition 4 has similarities with [13, Proposition 4.8] which was used to decide if a subgroup of $K^\times$ is p-saturated, and if not, to find an element of its p-saturation. However, the two statements are distinct, and neither implies the other.

Proposition 4. *Let $V = \langle U, \beta \rangle$ be finitely generated and assume $U \cap \langle \beta \rangle = \{1\}$ and that we are in the good case of Grunwald–Wang. Furthermore let $c \in \mathbb{R}_{>0}$ be arbitrary. Assume that the intersection*

$$\bigcap_{d \notin \mathfrak{p}, \mathrm{N}(\mathfrak{p}) \leq c} \ker(V/V^d \to \mathbb{Z}/d\mathbb{Z} \times k_\mathfrak{p}^\times/(k_\mathfrak{p}^\times)^d) \subseteq V/V^d$$

is generated by the classes of $\alpha_1\beta^{n_1}, \dots, \alpha_l\beta^{n_l} \in V$ with $\alpha_i \in U$, $n_i \in \mathbb{Z}$.

1. *If $\gcd(d, n_1, \dots, n_l) \neq 1$, then β is not a d-th power modulo U.*
2. *Assume β is not a d-th power modulo U. Then for c sufficiently large we have $\gcd(d, n_1, \dots, n_l) \neq 1$.*
3. *Assume β is a d-th power modulo U. Then for c sufficiently large we have $\gcd(d, n_1, \dots, n_l) = 1$ and that the element $\alpha_1^{k_1} \cdots \alpha_l^{k_l}\beta$ is a d-th power, where $k_i \in \mathbb{Z}$ are integers with $1 = k_0 d + \sum_{i=1}^{l} k_i n_i$.*

Proof. Let us denote by W/V^d the intersection of the kernels.

(1): Assume that β is a d-th power modulo U, that is, $\alpha\beta \in V \cap (K^\times)^d$ for some $\alpha \in U$. As $(V \cap (K^\times)^d)/V^d \subseteq W/V^d$, there exist integers $0 < k_i < d$ such that

$$\overline{\alpha\beta} = \overline{(\alpha_1\beta^{n_1})^{k_1} \cdots (\alpha_l\beta^{n_l})^{k_l}}$$

in $W/V^d \subseteq V/V^d$. As V is generated by U and β, the group V^d is generated by U^d and β^d. Hence there exists $\alpha_0 \in U$ and $k_0 \in \mathbb{Z}$ such that

$$\alpha\beta = (\alpha_1\beta^{n_1})^{k_1} \cdots (\alpha_l\beta^{n_l})^{k_l}\alpha_0^d(\beta^d)^{k_0}.$$

From $U \cap \langle\beta\rangle = \{1\}$ we get $1 = k_0 d + \sum_{i=1}^l k_i n_i$, i.e., $\gcd(d, n_1, \ldots, n_l) = 1$.

(2): Let c_0 be the constant from Proposition 3 and assume $c \geq c_0$. In particular it holds $(V \cap (K^\times)^d)/V^d = W/V^d$. Assume $\gcd(d, n_1, \ldots, n_l) = 1$. Then there exist $k_i \in \mathbb{Z}$, $0 \leq i \leq l$, such that $1 = k_0 d + \sum_{i=1}^l k_i n_i$. Then the element $\alpha = \alpha_1^{k_1} \cdots \alpha_l^{k_l}$ satisfies

$$\alpha\beta = \alpha\beta^{n_1k_1} \cdots \beta^{n_lk_l}\beta^{dk_0} = (\alpha_1\beta^{n_1})^{k_1} \cdots (\alpha_l\beta^{n_l})^{k_l}\beta^{dk_0},$$

that is $\overline{\alpha\beta} \in W/V^d = (V \cap (K^\times)^d)/V^d$ and β is a d-th power modulo U.

(3): Let c_0 be as in Proposition 3 and assume $c \geq c_0$. Note that as β is a d-th power modulo U, it follows from (1) that $\gcd(d, n_1, \ldots, n_l) = 1$. The result follows, since

$$\alpha^{k_1} \cdots \alpha^{k_l}\beta = (\alpha_1\beta^{n_1})^{k_1} \cdots (\alpha_l\beta^{n_l})^{k_l}(\beta^{k_0})^d$$

and for all $1 \leq i \leq l$ we have $\alpha_i\beta^{n_i} \in (K^\times)^d$ (as $c \geq c_0$). □

Remark 3. If $\mathfrak{p}$ is a prime ideal with $\gcd(d, \mathrm{N}(\mathfrak{p}) - 1) = 1$, then $k_\mathfrak{p}^\times = (k_\mathfrak{p}^\times)^d$. Thus we can always restrict to prime ideal $\mathfrak{p}$ with $\gcd(d, \mathrm{N}(\mathfrak{p}) - 1) \neq 1$. Additionally note that Theorem 5.2 already holds for a set T of prime ideals of Dirichlet density 1 with $T \cap S = \emptyset$ (see [34, Theorem 9.1.11]). As the set of prime ideals of $\mathcal{O}_K$ of degree 1 has Dirichlet density 1, Proposition 3 and Algorithm 2 remain correct if we consider almost all prime ideals of degree 1. This has two important consequences in practice:

1. By considering only prime ideals $\mathfrak{p}$ of degree 1, we always have $k_\mathfrak{p} \cong \mathbb{F}_p$, greatly simplifying the discrete logarithms that we have to compute.
2. We can skip finitely many prime ideals. For example, we can ignore those prime ideals, for which the reduction $V \to k_\mathfrak{p}$ is expensive to compute. If we represent K as $\mathbb{Q}(\alpha)$ with α integral, this means skipping prime ideals lying over rational primes which divide $[\mathcal{O}_K : \mathbb{Z}[\alpha]]$.

Algorithm 2 decides whether an element β is a a d-th power modulo U, and if so, it finds an element of u such that $u\beta$ is a d-th power. Proposition 4 directly shows its correctness. Algorithm 2 is an analogue of [13, Algorithm 4.9], (as well as the techniques from [8, Section 5.3], and more generally, the works derived from the number field sieve) in the sense that it uses *saturation techniques* to detect a d-th power. Kernels of a set of maps using information modulo various primes are used to produce candidates that are then tested in Step 7.

5.2. **The generic case.** We now assume that we are in the generic case and $d = p^r$ is a prime power. Since this includes the bad case of Grunwald–Wang, in general we cannot detect global d-th powers just using local information. The algorithms are therefore more complicated; the reader can skip this section without significantly affecting their understanding. We state results for an arbitrary p, but for us the only relevant case is $p = 2$. In general, we detect d-th powers for $d = 2^r d'$ with $2 \nmid d'$ by detecting d'-th powers using Section 5.1, detecting 2^r-th powers using results of

```
   Input  : $U \subseteq K^\times$ finitely generated, $\beta \in K^\times$ such that $U \cap \langle \beta \rangle = \{1\}$,
            $d = p^r$ a prime power, such that we are in the good case of
            Grunwald–Wang
   Output: Whether $\beta$ is a $d$-th power modulo $U$ and an element $\gamma \in K^\times$
            with $\beta/\gamma^d \in U$ in case it exists
 1 Let $c \in \mathbb{R}_{>0}$ (chosen arbitrarily);
 2 Determine a $(\mathbb{Z}/d\mathbb{Z})$-generating set $\overline{\alpha_1\beta^{n_1}}, \dots, \overline{\alpha_l\beta^{n_l}}$ of
       $\bigcap_{p \notin \mathfrak{p}, \mathrm{N}(\mathfrak{p}) \leq c} \ker(\langle U, \beta\rangle / \langle U, \beta\rangle^d \to \mathbb{Z}/d\mathbb{Z} \times k_\mathfrak{p}^\times/(k_\mathfrak{p}^\times)^d)$;
    if $\gcd(d, n_1, \dots, n_l) \neq 1$ then
 3      return: $\beta$ is not a $d$-th power modulo $U$;
 4 else if $\gcd(d, n_1, \dots, n_l) = 1$ then
 5      Determine $k, k_i \in \mathbb{Z}$, $1 \leq i \leq l$, with $1 = kd + \sum_{i=1}^l k_i n_i$;
 6      Test whether the element $\delta = \alpha_1^{k_1} \cdots \alpha_l^{k_l}$ is a $d$-th power;
 7      if there exists $\gamma$ with $\gamma^d = \delta$ then
 8          return: $\gamma$;
 9      end
10 Replace $c$ by $2c$ and go to step 2;
```

Algorithm 2: d-th power modulo units in the good case

this section, and recombining the results using Lemma 5.1. To still be able to use the technique of Section 5.1, we investigate the situation where U is p-saturated.

Definition 5.3. The *p-saturation* V of U is the smallest subgroup $V \subseteq K^\times$ with $U \subseteq V$ and $K^\times/V$ p-torsion-free. The group U is called *p-saturated* or *saturated at p* if U equals its p-saturation, that is, $K^\times/U$ is p-torsion-free. The group U is *saturated* if it is p-saturated for all primes p.

Under the GRH, when $c > c_0 = 72d^2(\log|\Delta_K| + 3n\log(p))^2$, Algorithm 4.9 of [13] correctly returns the p-saturation of the input subgroup U of the unit group in polynomial time [13, Th. 4.11]. We now assume that U is p-saturated, and we show that testing whether β is a p^r-th power modulo U can be reduced to r instances of the problem where the exponent is p (instead of p^r), hence to a situation where we are in the good case of Grunwald–Wang.

Proposition 5. *Assume that $U \subseteq K^\times$ is a multiplicative group and $\beta \in K^\times$ a p^r-th power modulo U. Then the following hold:*

1. *The element β is a p^i-th power modulo U for all $1 \leq i \leq r$.*
2. *Assume that U is p-saturated and that there exist $u \in U$, $\gamma_i \in K^\times$ with $u\beta = \gamma_i^{p^i}$ for some $1 \leq i \leq r-1$. Then γ_i is a p^{r-i}-th power modulo U.*

Proof. (1): Trivial. For (2), first note that by assumption there exist $\tilde{u} \in U$, $\gamma \in K^\times$ such that $\tilde{u}\beta = \gamma^{p^r}$. Then

$$\gamma_i^{p^i} = u\beta = \frac{u}{\tilde{u}}\tilde{u}\beta = \frac{u}{\tilde{u}}\gamma^{p^r}.$$

Hence

$$\frac{u}{\tilde{u}} = \frac{\gamma_i^{p^i}}{\gamma^{p^r}} = \left(\frac{\gamma_i}{\gamma^{p^{r-i}}}\right)^{p^i} \in U.$$

As U is p-saturated this implies $\gamma_i/\gamma^{p^{r-i}} \in U$. Thus $(\gamma^{p^{r-i}}/\gamma_i)\gamma_i = \gamma^{p^{r-i}}$ shows that γ_i is a p^{r-i}-th power modulo U. □

Corollary 1. *Assume that $U \subseteq K^\times$ is a p-saturated multiplicative group. An element $\beta \in K^\times$ is a p^r-th power modulo U if and only if there exist $u_1, \ldots, u_{r-1} \in U$, $\gamma_1, \ldots, \gamma_r \in K^\times$, $\gamma_1 = \beta$ such that $\gamma_{i+1}^p = u_i\gamma_i$ for all $1 \leq i \leq r-1$.*

Therefore, under the assumption that U is p-saturated, we can check whether β is a p^r-th power modulo U by iteratively checking whether certain elements are p-th powers modulo U. As p is a prime, we are always in the good case of the Grunwald–Wang theorem and we can use the technique of Section 5.1. We summarize this in Algorithm 3, which is correct according to Corollary 1.

```
Input  : A $p$-saturated multiplicative group $U \subseteq K^\times$, $r \geq 1$, and $\beta \in K^\times$
         with $U \cap \langle \beta \rangle = \{1\}$
Output: Whether $\beta$ is a $p^r$-th power modulo $U$ and an element $\gamma \in K^\times$
         with $\beta/\gamma^{p^r} \in U$ in case it exists
$\gamma_0 \leftarrow \beta$;
for $i \leftarrow 1$ to $r$ do
    if $\gamma_{i-1}$ is a $p$-th power modulo $U$ using Algorithm 2 then
        Compute $\gamma_i \in K^\times$ such that $\gamma_{i-1}/\gamma_i^p \in U$;
    else
        return: that $\beta$ is not a $p^r$-th power modulo $U$;
    end
end
return: $\gamma_r$;
```

Algorithm 3: d-th power in the bad case

5.3. **Computing roots.** An important subproblem of the previous section is the computation of roots. More precisely, given $\delta \in K$ and $d \in \mathbb{Z}_{>0}$, we need to decide whether there exists an element $\gamma \in K$ such that $\gamma^d = \delta$ (and if so, then compute γ). Since δ will in general be quite large, we first compute a compact representation with respect to d, which amounts to finding small elements $\delta_0, \ldots, \delta_k \in K$ such that

$$\delta = \delta_0 \cdot \delta_1^d \cdot \cdots \cdot \delta_k^{d^k}.$$

This presentation was introduced by Thiel [40] for units. An algorithm for finding such a presentation for S-units goes back to lecture notes of Fieker and has subsequently been used in [9, 10, 14, 24]. Given such a presentation, it is clear that δ is a d-th power if and only if δ_0 is a d-th power. For the latter task, we use Hensel lifting of the linear factors of $X^d - \beta_0 \in K[X]$ modulo a non-zero prime ideal $\mathfrak{p}$ of $\mathcal{O}_K$, as described in [22]. Note that for both the computation of the compact representation and the root computation, the main computational task is the computation of LLL bases of ideals of $\mathcal{O}_K$, that is, lattice reduction of lattices of dimension $[K : \mathbb{Q}]$. To evaluate the performance of the compact representation, we need a notion of size of an element in K.

Definition 5.4. Let K be a number field whose complex embeddings are denoted by $\sigma_1, \ldots, \sigma_r, \sigma_{r+1}, \bar{\sigma}_{r+1}, \ldots, \sigma_{r+s}, \bar{\sigma}_{r+s}$. We define the T_2-norm of α to be $\|\alpha\| = \left(\sum_{1 \leq i \leq r+2s} |\sigma_i(\alpha)|^2\right)^{1/2}$.

The value $\|\alpha\|$ is a good measure of the size of an element of α. Indeed, as recalled in [12, Sec. 3], the maximum absolute value of a coefficient of $\alpha \in \mathcal{O}_K$ when represented on an LLL-reduced integral basis is less than $2^{3n/2}\|\alpha\|$, and, when $\alpha = \alpha_0/d$ for $d \in \mathbb{Z}_{>0}$ and $\alpha_0 \in \mathcal{O}_K$, the bit size $\mathrm{S}(\alpha)$ of α is less than $n\left(\frac{3n}{2}\log\|\alpha\| + \log(d)\right)$.

Proposition 6. *Let $\alpha = \prod_{i\leq l}\alpha_i^{x_i} \in \mathcal{O}_K$ such that we have $\max_i(\mathrm{S}(\alpha_i)) \leq B_\alpha$, and $\max_i(\log|x_i|) \leq B_x$. There exists an algorithm for computing a compact representation of α in time*

$$\mathrm{Poly}(\log|\Delta_K|, B_x, B_\alpha) + \mathrm{Fact}(\mathrm{N}(\alpha)),$$

where $\mathrm{Fact}(\mathrm{N}(\alpha))$ denotes the cost of factoring the norm of α. When this factorization is known, this cost is 0.

Subfield unit calculations, and subfield resolutions of the PIP are assumed to be followed by a compact representation routine. In both cases, the prime factorization of the input is known in advance, therefore $\mathrm{Fact}(\mathrm{N}(\alpha)) = 0$. Moreover, the product of a polynomial number of terms in compact representation can be kept in a compact representation by direct multiplication of the terms. Therefore, the compact representation algorithm is only executed in subfields. Then, operations on compact representations in field extensions have polynomial run time.

We conclude this section with the resulting complexity of Algorithm 1.

Theorem 5.5 (under GRH)**.** *Algorithm 1 is correct and has complexity*

$$\mathrm{Poly}([K:\mathbb{Q}], \log|\Delta_K|, \log(\mathrm{N}(\mathfrak{a})), l, \max_i \log(a_i)) + l \cdot \mathrm{PIP}(\mathrm{Subfields}),$$

where $\mathrm{PIP}(\mathrm{Subfields})$ denotes the cost of Step (3) (PIP in a subfield).

Proof. From Lemma 5.1 it follows that it is sufficient to show that Algorithms 2 and 3 have the claimed complexity. We first consider Algorithm 2. From [13, Theorem 4.11] it follows that the algorithm terminates as soon as $c > 72d^2(\log|\Delta_K| + 3n\log(d))^2$, hence after a number of steps which is polynomial in the size of the input. As the final root computation has runtime polynomial in the size of the input, this proves the claim for Algorithm 2.

For Algorithm 3, first note that a 2-saturated subgroup $U \subseteq \mathcal{O}_K^\times$ can be computed in polynomial time ([13, Corollary 4.13]). As the successive applications of Algorithm 2 for $p = 2$ have the same complexity, the claim follows. □

6. **Fast asymptotic performance in certain fields.** Using the heuristics methods of Biasse and Fieker [5, 11], the heuristic asymptotic run time of solving the PIP on input $\mathfrak{a}$ in a number field K of discriminant Δ is in $\mathrm{Poly}(\log(\mathrm{N}(\mathfrak{a})))\cdot 2^{\tilde{O}((\log|\Delta|)^{2/3})}$. In this section, we briefly illustrate the potential asymptotic gains of solving the PIP with subfields by exhibiting an infinite family of cyclotomic fields $(K_k = \mathbb{Q}(\zeta_{m_k}))_{k\geq 1}$ where our algorithm has heuristic-free cost $\mathrm{Poly}(\log(\mathrm{N}(\mathfrak{a})))\cdot 2^{(\log|\Delta|)^{o(1)}}$.

We consider a number field K with abelian Galois group G. Recall from Theorem 3.1 that when $G \simeq C \times Q$, where C is the largest cyclic factor, and $|Q|$ is divisible by 2 distinct primes, we have a norm relation where the subfields have degree bounded by $|C|$. The run time is therefore minimized in families of fields where the largest cyclic factor C of G is as small as possible. In the case of a cyclotomic field $\mathbb{Q}(\zeta_m)$ we have $G = (\mathbb{Z}/m\mathbb{Z})^\times$, $|C| = \lambda(m)$, where λ is the the Carmichael function, while $[\mathbb{Q}(\zeta_m) : \mathbb{Q}] = \varphi(m)$. We can construct an infinite family of conductors with small Carmichael numbers by using the following theorem of Erdös, Pomerance and Schmutz.

Theorem 6.1 (Erdös–Pomerance–Schmutz [21, Theorem 1 part 2])**.** *There exists an infinite sequence $m_1 < m_2 < \dots$ of positive integers such that*

$$\lambda(m_k) = (\log(m_k))^{O(\log\log\log(m_k))}.$$

Remark 4. Integers as in Theorem 6.1 can easily be constructed in practice as follows. Let L be a highly divisible number (for instance, take L to be a product of a few small primes). Then let Q be the set of all primes p such that $p-1$ divides L, and let $m = \prod_{p\in Q} p$. This integer satisfies $\lambda(m) \mid L$, and the proof of Theorem 6.1 shows that for suitable choices of L, the integer m is much larger than L.

Example 2. We illustrate the construction by taking L to be the product of the first prime numbers.

a) $L = 2\cdot 3 = 6$, $m = 2\cdot 3\cdot 7 = 42$, $\varphi(m) = 12$, $\lambda(m) = 6$.
b) $L = 2\cdot 3\cdot 5 = 30$, $m = 2\cdot 3\cdot 7\cdot 11\cdot 31 = 14322$, $\varphi(m) = 3600$, $\lambda(m) = 30$.
c) $L = 2\cdot 3\cdot 5\cdot 7 = 210$, $m = 2\cdot 3\cdot 7\cdot 11\cdot 31\cdot 43\cdot 71\cdot 211 = 9225988926$, $\varphi(m) = 2222640000$, $\lambda(m) = 210$.

In order to obtain a heuristic-free complexity result, we use the following weak bound on the complexity of the computation of class and unit groups in number fields due to Lenstra.

Theorem 6.2 (Lenstra [31, Th. 5.5])**.** *One can determine the class and unit group of a number field K in deterministic time $2^{(\log|\Delta_K|)^{O(1)}}$.*

We obtain the following heuristic-free complexity result.

Theorem 6.3 (under GRH)**.** *There exists an infinite sequence of integers $m_1 < m_2 < \dots$ such that Algorithm 1 applied to the family of fields $K_k = \mathbb{Q}(\zeta_{m_k})$ with input $\mathfrak{a} \subseteq \mathcal{O}_{K_k}$ has complexity*

$$\mathrm{Poly}([K_k : \mathbb{Q}], \log(\mathrm{N}(\mathfrak{a}))) \cdot 2^{(\log(m_k))^{O(\log\log\log(m_k))}}.$$

Proof. Take $(m_k)_{k\in\mathbb{N}}$ to be the sequence from Theorem 6.1. Let $m = m_k$ be a term in this sequence, and $K_k = \mathbb{Q}(\zeta_{m_k})$ be the corresponding field. Let D be the maximum absolute value of the discriminant of a subfield used by the subexponential PIP algorithm of [5, 11] applied to K_k. Then we have $D \leq m^{\lambda(m)}$, so that

$$\log(D) \leq \lambda(m)\log(m) = (\log(m))^{O(\log\log\log(m))}$$

by Theorem 6.1. In particular, using the algorithm of Theorem 6.2 for the base case, the cost for the subfields is $2^{(\log(D))^{O(1)}} = 2^{(\log(m))^{O(\log\log\log(m))}}$ □

Remark 5. Let Δ_k be the discriminant of K_k. Then we have $\log(m_k) = O(\log\log|\Delta_k|)$, so that the second term of the complexity is

$$2^{(\log\log|\Delta_k|)^{O(\log\log\log\log|\Delta_k|)}}.$$

This complexity is not quite quasi-polynomial (which would correspond to $O(1)$ instead of $O(\log\log\log\log|\Delta_k|)$ in the second exponent), but it is strongly subexponential, as can be seen by rewriting it as

$$2^{(\log|\Delta_k|)^{O\left(\frac{\log\log\log\log|\Delta_k|\,\log\log\log|\Delta_k|}{\log\log|\Delta_k|}\right)}} = 2^{(\log|\Delta_k|)^{o(1)}}.$$

This complexity would not be significantly improved by using the best known heuristic algorithms [10] for the PIP and ideal decomposition instead of Theorem 6.2, as it would only improve the implicit constant in the big O.

7. **Numerical results.** We implemented our algorithm using the algebra package Hecke [23] (written in Julia [4]). We used 55 nodes of 20 cores 2x Intel(R) Xeon(R) Silver 4114 CPU @ 2.20GHz processors with 192GB memory. We focused our attention on examples outside of the reach of the previous techniques by a substantial margin: The field $K^{(1)} = \mathbb{Q}(\zeta_{825})$ of degree 400 and discriminant $\approx 10^{960}$, the field $K^{(2)} = \mathbb{Q}(\zeta_{3276})$ of degree 864 and discriminant $\approx 10^{2369}$, and the field $K^{(3)} = \mathbb{Q}(\zeta_{2387})$ of degree 1800 and discriminant $\approx 10^{5539}$. For each field, we report on the resolution of one instance of (S)PIP chosen at random. The class groups of the fields $K^{(1)}$ and $K^{(2)}$ have been determined using norm relations in [13]. That computation took less than two hours using a single core, but note that the method employed for the class group computations avoids taking the roots of elements and is therefore computationally much easier than solving the PIP. In particular, it cannot be used to compute S-unit groups or to solve the PIP.

Norm relations. We used a two step approach which is rigorously analyzed in [13, Th. 2.27] in the abelian case. First, we try to choose a norm relation of denominator 1 of the form $x = \prod_{i=1}^{l} \mathrm{N}_{K/K_i}(x^{b_i})^{a_i}$ such that the quantity $\max_{1\leq i\leq l}[K_i : \mathbb{Q}]$ is minimal. In other words, we try to find a norm relation of denominator 1 where the degrees of the subfields are as small as possible. As a second step, for each subfield K_i, we try to find a norm relation $x^{d_i} = \prod_{j=1}^{l_i} \mathrm{N}_{K_i/K_{i,j}}(x^{b_{i,j}})^{a_{i,j}}$ with d_i as small as possible, such that $\max_{1\leq j\leq l_i}[K_{i,j} : \mathbb{Q}]$ is bounded by some heuristically chosen constant B. To test whether some fractional ideal $\mathfrak{a}$ of K is principal, Algorithm 1 is now applied first using the norm relation of K of denominator 1 and then again using the norm relations of the K_i when testing whether $\mathrm{N}_{K/K_i}(\mathfrak{a}_i^{b_i})$ is principal. In particular this means that:

1. The largest degree of a field where we have to compute roots modulo units is $\max_{1\leq i\leq l}[K_i : \mathbb{Q}]$.
2. The largest degree of a field where we have to classically solve the principal ideal problem and compute (a saturated subgroup of) the unit group is bounded by B.

Since the norm relations are too large to display, we present several values quantifying the difficulty of solving the PIP in Table 1.

Table 1. Quantification of hardness of instances

K	$[K : \mathbb{Q}]$	$l = \#\{K_i\}$	$n = \max_i[K_i : \mathbb{Q}]$	$\#\{K_{i,j}\}$	$m = \max_{i,j}[K_{i,j} : \mathbb{Q}]$
$K^{(1)}$	400	19	100	86	20
$K^{(2)}$	864	38	108	341	12
$K^{(3)}$	1800	131	150	297	30

Recall that n is the maximal degree of the subfields where saturation and root computation needs to take place. Likewise, m is the maximal degree of a subfield where the PIP must be solved with a subexponential method and the column labeled "$\#\{K_{i,j}\}$" denotes the number of these subfields. In particular, we observe

that the saturation and root computation to solve the PIP in the field $K^{(3)}$ of degree 1800 only occurs in fields of degree bounded by 150 while the subexponential computations occur in subfields of degree no more than 30.
Results. We ran the computation of subexponential PIP instances in the subfields on independent cores. A second layer of parallelization was employed by computing individual roots on independent cores. After picking a principal ideal $\mathfrak{a} = (\alpha)$ of $\mathcal{O}_K$, the main steps of our computations are the following: (1) Finding the initial norm relation to determine the subfields K_i, (2) Finding the norm relation in each of the subfields K_i, (3) computing the subfields $K_{i,j}$, (4) Computing the unit groups of the $K_{i,j}$, (5) Computing the relative norms $\mathrm{N}_{K/K_{i,j}}(\mathfrak{a})$, (6) Computing generators of the ideals $\mathrm{N}_{K/K_{i,j}}(\mathfrak{a})$, (7) Identifying d-powers (without root computation), (8) Compact representation, and (9) Root computation.

Table 2. PIP Runtime in CPU hours of Steps 1 to 9.

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	Total
$K^{(1)}$	0.46	2.70	0.33	0.25	0.55	0.13	0.05	0.37	1.71	6.55
	7.0%	41.2%	5.0%	3.8%	8.4%	2.0%	0.8%	5.6%	26.1%	
$K^{(2)}$	4.12	15.23	2.90	1.00	11.21	1.54	0.12	2.61	60.19	98.92
	4.2%	15.4%	3.0%	1.0%	11.3%	1.6%	0.1%	2.6%	60.8%	
$K^{(3)}$	66.61	140.00	58.34	102.15	641.92	203.62	3.25	55.19	1634.01	2905.09
	2.3%	4.8%	2.0%	3.5%	22.1%	7.0%	0.1%	1.9%	56.2%	

We also implemented the reduction from the SPIP to the PIP of [19, 20] and we were able to retrieve a short generator of our challenge ideals in $K^{(1)}$ and $K^{(2)}$ (which is a solution to γ-SVP for a $\gamma \in e^{\tilde{O}(\sqrt{n})}$ in the input principal ideals). This incurred an additional 6.9min of CPU time for $K^{(1)}$, and 3.3h for $K^{(2)}$.

8. **Comparison with the S-unit method.** In this section, we provide heuristic arguments, as well as numerical evidence, to illustrate the fact that Algorithm 1 performs better in practice than the PIP resolution based on the recursive computation of S-units from norm relations sketched in Section 2, and described more formally in Algorithm 4.

Algorithm 1 was designed with the practical performance in mind. This is why we strive to avoid the compact representation and saturation steps as much as possible through the contributions described in Section 4 and Section 5. In particular, the units used are directly coming from subfields, and seldom need saturation. On the other hand, computing S-units as in Algorithm 4 would require significantly more compact representation and saturation steps, and the compact representation algorithm would receive inputs divisible by conjugates of $\mathfrak{p}$, which is a large ideal. Let us provide coarse estimates of this slow down.

At the beginning, we can assume that we have performed an LLL-reduction on the input and replaced $\mathfrak{a}$ by $(b_1)/\mathfrak{a}$ where b_1 is the first basis vector. This means that we may assume

$$\mathrm{N}(\mathfrak{a}) \leq \lambda^n \sqrt{|\Delta|} \in O\left(2^{O(n^2)}\sqrt{|\Delta|}\right),$$

with $n = \deg(K)$, and $\lambda \sim 2^{O(n)}$ the approximation factor of LLL. Let $b_1, \ldots, b_n$ be an LLL-reduced basis of $\mathfrak{a}$, so that

$$\|b_1\| \leq \lambda |\Delta|^{1/2n} \mathrm{N}(\mathfrak{a})^{1/n} \in 2^{O(n)} |\Delta|^{1/2n} \mathrm{N}(\mathfrak{a})^{1/n}.$$

```
Input  : $\mathfrak{a} \subseteq \mathcal{O}_K$ principal
Output: A generator $g \in \mathcal{O}_K$ of $\mathfrak{a}$
$B \leftarrow$ LLL-reduced basis of $\mathfrak{a}$;
$\alpha \xleftarrow{\mathcal{R}} \mathrm{Span}(B)$.// $\alpha$ chosen uniformly at random;
while $\mathfrak{p} = (\alpha)/\mathfrak{a}$ is not prime do
    $\alpha \xleftarrow{\mathcal{R}} \mathrm{Span}(B)$;
end
$S \leftarrow \{\mathfrak{p}^\sigma$ for $\sigma \in \mathrm{Gal}(K/\mathbb{Q})\}$;
Find generators $(\alpha_i)_{s+r}$ of the $S$-unit group using [13, Alg. 4.16];
Let $M \in \mathbb{Z}^{(r+s)\times s}$ such that row $i$ is the valuations of $\alpha_i$;
Solve $\vec{x} \cdot M = \vec{y}$ for $\vec{y} = (1, 0, \dots, 0)$;
return: $\alpha \cdot \prod_i \alpha_i^{-x_i}$;
```

Algorithm 4: Solving the PIP using S-units

This means that $\mathrm{N}(b_1) \in 2^{O(n^2)}\sqrt{|\Delta|}\mathrm{N}(\mathfrak{a})$ and

$$\mathrm{N}\left((b_1)/\mathfrak{a}\right) \in 2^{O(n^2)}\sqrt{|\Delta|}$$

Due to the density of prime numbers, the number of times we expect to need to draw an element of norm $2^{O(n^2)}\sqrt{|\Delta|}$ before finding one whose norm is prime is about $O(n^2)$. The chosen strategy for enumeration of short elements in $\mathfrak{a}$ is to draw elements of the form

$$\alpha = b_{i_1} + b_{i_2} + \dots, +b_{i_c},$$

for a constant c (typically $c = 3$ to ensure that the search space is large enough to find a prime norm) and a choice of c random indices $i_1, \cdots, i_c$. The vectors $b_2, \dots, b_n$ are at least as long as b_1, but in the best case scenario, all vectors are of the same length. If this were the case, then $\|\alpha\| \sim \sqrt{c}\|b_1\|$. In practice, we expect $\|\alpha\|$ to be in fact larger. In any case, the algebraic norm of α is expected to satisfy

$$\mathrm{N}\left(\frac{(\alpha)}{\mathfrak{a}}\right) \geq c^{n/2}\mathrm{N}\left(\frac{(b_1)}{\mathfrak{a}}\right) \sim c^{n/2}\lambda^n\sqrt{|\Delta|} \sim c^{n/2}\mathrm{N}(\mathfrak{a}).$$

Hence the bit size of the norm of an S-unit where $S = \{\mathfrak{p}^\sigma \mid \mathfrak{p} = (\alpha)/\mathfrak{a}, \sigma \in \mathrm{Gal}(K/\mathbb{Q})\}$ is expected to be $O(n)$ times larger than the bit size of the norm of the generator of $\mathfrak{a}$. In addition, the cardinality of S is equal $|\mathrm{Gal}(K/\mathbb{Q})| + r \in O(n)$ where r is the rank of the unit group of K. Here we assume that $\mathfrak{p}$ is of degree 1, which happens the majority of the time. In any case, $r \in \Omega(n)$ is a lower bound on $|S|$.

In the compact representation algorithm, the complexity of the calls to LLL is proportional to the $\log(\mathrm{N}(\mathfrak{a}'))$ where $\mathfrak{a}'$ is an ideal whose norm is proportional to $\prod_{\mathfrak{p}} \mathrm{N}(\mathfrak{p})$ where $\mathfrak{p}$ runs over the prime ideals dividing the input element. Hence, when computing compact representations of S-units for S defined above, the execution time of the compact representation algorithm is proportional to $|S|\log(\mathrm{N}(\mathfrak{p}))$ where $\mathfrak{p} = (\alpha)/\mathfrak{a}$. On the other hand, this term becomes $\log(\mathrm{N}(\mathfrak{a}))$ when the compact representation is called on a generator of $\mathfrak{a}$. Therefore, each call to the compact representation on input an S-unit is expected to be $O(n^2)$ times more expensive than the call on input a generator of $\mathfrak{a}$. Moreover, to solve the PIP with Algorithm 4, we need the entire S-unit group, which means that there needs to be $|S|$ calls to the compact representation instead of a single one on the generator of $\mathfrak{a}$, hence multiplying the compact representation effort by an $O(n)$ factor. Altogether, we estimate that the S-unit based resolution of the PIP should be $O(n^3)$ times slower than the methods introduced in this paper.

Asymptotically, the slowdown induced by opting for Algorithm 4 does not impact the overall complexity which is strongly subexponential. However, given that this complexity is somewhat close to being polynomial, an n^3 slowdown does impact concrete computations to the point that the large degree calculations presented in this paper are infeasible with the S-unit method. To illustrate the sharp increase of the slowdown, we compared the two methods for small instances of increasing difficulty on a single core Intel(R) Xeon(R) Silver 4114 CPU @ 2.20GHz with 192GB memory. The source code for these tests is supplied with this submission. We see in Table 3 that trivial examples are more easily solved by using S-units because in small dimension, the random choice of a small element in $\mathfrak{a}$ almost always directly yields a generator. However, the method described in this paper is already showing its impact on examples taking 30 min (by being twice as fast), and on examples taking several hours on a single core, it is already faster by a significant margin. This sharp increase backs our heuristic estimate of an n^3 speedup. We further document our estimate by running the following experiment. In the degree 1800 field $K^{(3)}$ of Section 7, the chosen PIP challenge has the prime decomposition $\mathfrak{a} = \mathfrak{p}_1\mathfrak{p}_2$ where $\mathrm{N}(\mathfrak{p}_1) = 133673$ and $\mathrm{N}(\mathfrak{p}_2)$ is a 733 bit prime. On the other hand, our experiments showed that random short elements $\alpha \in \mathfrak{a}$ had algebraic norms of about 1,000,000 bits. This is consistent with our estimate that the bit size of the norm of the primes in S is $O(n)$ times larger than the bit size of the norm of $\mathfrak{a}$.

Table 3. Comparison with the S-unit method

$[K : \mathbb{Q}]$	This paper	S-unit method
36	5.5sec	0.3 sec
72	2.1 min	1.6 min
144	31.7 min	1.1h
216	3.6h	54h

9. **Conclusion and future work.** We have described a PIP algorithm that uses the norm relations of [13] that offers better practical performances than prior works (including those that explicitely or implicitely use norm relations). We were able to solve the PIP in a field of degree 1800, and to find short generators of ideals in a field of degree 864.

Short generators of principal ideals are of cryptographic interest because they are solutions to the γ-SVP for $\gamma \in 2^{\tilde{O}(\sqrt{n})}$. Such solutions are sometimes referred to as *mildly short* vectors. The search for mildly short vectors in non-principal ideals was shown to reduce to the PIP in [20]. The heuristic efficient reduction of [20] is quantum and relies on the decomposition of classes of ideals with respect to a set of generators of the class group. The recursive S-unit algorithm of [13] can be used to perform the reduction of γ-SVP to PIP of [20]. The resulting algorithm enjoys the same asymptotic complexity as the methods presented in this paper. In particular, in special families of fields such as those described in Section 6, the norm relation method has strong subexponential heuristic complexity, which is a superpolynomial improvement over the complexity of BKZ. However, the practical implications of this norm-relation approach are not clear. Indeed, for most inputs, the full S-unit group would need to be calculated for some large set S, thus suffering from the same practical limitations as the S-unit based PIP method described in Section 8.

REFERENCES

[1] L. M. Adleman, Factoring numbers using singular integers, in *Proceedings of STOC '91*, 1991, 64–71.

[2] E. Artin and J. Tate, *Class field theory*, AMS Chelsea Publishing, Providence, RI, 2009, Reprinted with corrections from the 1967 original.

[3] J. Bauch, D. Bernstein, H. de Valence, T. Lange and C. van Vredendaal, Short generators without quantum computers: The case of multiquadratics, in *Proceedings of EUROCRYPT 2017*, 2017, 27–59, URL https://doi.org/10.1007/978-3-319-56620-7_2.

[4] J. Bezanson, A. Edelman, S. Karpinski and V. Shah, Julia: A fresh approach to numerical computing, *SIAM review*, **59** (2017), 65–98, URL https://doi.org/10.1137/141000671.

[5] J.-F. Biasse, Subexponential time relations in the class group of large degree number fields, *Advances in Mathematics of Communications*, **8** (2014), 407–425, URL http://aimsciences.org/journals/displayArticlesnew.jsp?paperID=10551.

[6] J.-F. Biasse, Approximate short vectors in ideal lattices of $\mathbb{Q}(\zeta_{p^e})$ with precomputation of $\mathrm{Cl}(\mathcal{O}_k)$, in *Proceedings of SAC 2017*, 2018, 374–393.

[7] J. Biasse, T. Espitau, P. Fouque, A. Gélin and P. Kirchner, Computing generator in cyclotomic integer rings, in *Proceedings of EUROCRYPT 2017*, 2017, 60–88.

[8] J.-F. Biasse and C. Fieker, New techniques for computing the ideal class group and a system of fundamental units in number fields, *CoRR*, **abs/1204.1294**.

[9] J.-F. Biasse and C. Fieker, Improved techniques for computing the ideal class group and a system of fundamental units in number fields, in *Proceedings of ANTS X*, 2013, 113–133.

[10] J.-F. Biasse and C. Fieker, Subexponential class group and unit group computation in large degree number fields, *LMS J. Comput. Math.*, **17** (2014), 385–403, URL https://doi.org/10.1112/S1461157014000345.

[11] J.-F. Biasse and C. Fieker, Subexponential class group and unit group computation in large degree number fields, *LMS Journal of Computation and Mathematics*, **17** (2014), 385–403, URL http://journals.cambridge.org/article_S1461157014000345.

[12] J. Biasse, C. Fieker and T. Hofmann, On the computation of the HNF of a module over the ring of integers of a number field, *J. Symb. Comput.*, **80** (2017), 581–615, URL https://doi.org/10.1016/j.jsc.2016.07.027.

[13] J.-F. Biasse, C. Fieker, T. Hofmann and A. Page, Norm relations and computational problems in number fields, 2020, ArXiv:2002.12332.

[14] J.-F. Biasse and F. Song, Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields, in *Proceedings of SODA16*, 2016, 893–902.

[15] J.-F. Biasse and C. van Vredendaal, Fast multiquadratic S-unit computation and application to the calculation of class groups, in *Proceedings of ANTS XIII*, 2019, 103–118.

[16] J. Buchmann, A subexponential algorithm for the determination of class groups and regulators of algebraic number fields, in *Séminaire de Théorie des Nombres*, Paris, 1989, 27–41.

[17] P. Campbell, M. Groves and D. Shepherd, SOLILOQUY: a cautionary tale, Online draft available at http://docbox.etsi.org/Workshop/2014/201410_CRYPTO/S07_Systems_and_Attacks/S07_Groves_Annex.pdf.

[18] H. Cohen, F. D. Y. Diaz and M. Olivier, Subexponential algorithms for class group and unit computations, *Journal of Symbolic Computation*, **24** (1997), 433 – 441, URL http://www.sciencedirect.com/science/article/pii/S0747717196901431.

[19] R. Cramer, L. Ducas, C. Peikert and O. Regev, Recovering short generators of principal ideals in cyclotomic rings, in *Proceedings of EUROCRYPT 2016*, 2016, 559–585.

[20] R. Cramer, L. Ducas and B. Wesolowski, Short Stickelberger class relations and application to Ideal-SVP, in *Proceedings of EUROCRYPT 2017*, 2017, 324–348.

[21] P. Erdős, C. Pomerance and E. Schmutz, Carmichael's lambda function, *Acta Arith.*, **58** (1991), 363–385.

[22] C. Fieker and C. Friedrichs, On reconstruction of algebraic numbers, in *Algorithmic number theory (Leiden, 2000)*, vol. 1838 of Lecture Notes in Comput. Sci., Springer, Berlin, 2000, 285–296, URL https://doi.org/10.1007/10722028_16.

[23] C. Fieker, W. Hart, T. Hofmann and F. Johansson, Nemo/Hecke: Computer algebra and number theory packages for the Julia programming language, in *Proceedings of ISSAC 2017*, 2017, 157–164.

[24] C. Fieker, T. Hofmann and C. Sircana, On the construction of class fields, in *Proceedings of ANTS XIII*, 2019, 239–255.

[25] T. Funakura, On Artin theorem of induced characters, *Comment. Math. Univ. St. Paul.*, **27** (1978/79), 51–58.

[26] S. Garg, C. Gentry and S. Halevi, Candidate multilinear maps from ideal lattices, in *Proceedings of EUROCRYPT 2013*, 2013, 1–17.

[27] C. Gentry and M. Szydlo, Cryptanalysis of the revised NTRU signature scheme, in *Proceedings of EUROCRYPT 2002*, 2002, 299–320.

[28] J. Hafner and K. McCurley, A rigorous subexponential algorithm for computation of class groups, *Journal of American Mathematical Society*, **2** (1989), 839–850.

[29] M. Kirschmer, *Definite quadratic and hermitian forms with small class number*, Habilitationsschrift, RWTH Aachen University, 2016.

[30] A. Lenstra, H. Lenstra and L. Lovász, Factoring polynomials with rational coefficients, *Mathematische Annalen*, **261** (1982), 515–534.

[31] H. W. Lenstra Jr., Algorithms in algebraic number theory, *Bull. Amer. Math. Soc. (N.S.)*, **26** (1992), 211–244.

[32] A. Lesavourey, T. Plantard and W. Susilo, Short Principal Ideal Problem in multicubic fields, *J. Math. Cryptol.*, **14** (2020), 359–392.

[33] J. Neukirch, *Algebraic number theory*, Comprehensive Studies in Mathematics, Springer-Verlag, 1999, ISBN 3-540-65399-6.

[34] J. Neukirch, A. Schmidt and K. Wingberg, *Cohomology of number fields*, vol. 323 of Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences], 2nd edition, Springer-Verlag, Berlin, 2008, URL https://doi.org/10.1007/978-3-540-37889-1.

[35] O. T. O'Meara, *Introduction to quadratic forms*, Die Grundlehren der mathematischen Wissenschaften, Bd. 117, Academic Press, Inc., Publishers, New York; Springer-Verlag, Berlin-Göttingen-Heidelberg, 1963.

[36] M. Pohst and H. Zassenhaus, *Algorithmic algebraic number theory*, vol. 30 of Encyclopedia of Mathematics and its Applications, Cambridge University Press, Cambridge, 1989, URL https://doi.org/10.1017/CBO9780511661952.

[37] C. P. Schnorr and M. Euchner, Lattice basis reduction: Improved practical algorithms and solving subset sum problems, *Math. Program.*, **66** (1994), 181–199, URL http://dx.doi.org/10.1007/BF01581144.

[38] D. Simon, *Équations dans les corps de nombres et discriminants minimaux*, PhD thesis, Université Bordeaux I, 1998.

[39] N. Smart and F. Vercauteren, Fully homomorphic encryption with relatively small key and ciphertext sizes, in *Proceedings of PKC 2010*, 2010, 420–443.

[40] C. Thiel, *On the complexity of some problems in algorithmic algebraic number theory*, PhD thesis, Universität des Saarlandes, 1995.

Received xxxx 20xx; revised xxxx 20xx.

E-mail address: biasse@usf.edu
E-mail address: fieker@mathematik.uni-kl.de
E-mail address: tommy.hofmann@uni-siegen.de
E-mail address: wyoumans@usf.edu