On the Infrastructure of the Principal Ideal Class of an Algebraic Number Field of Unit Rank One

By Johannes Buchmann* and H. C. Williams**

Abstract. Let $R$ be the regulator and let $D$ be the absolute value of the discriminant of an order $\mathcal{O}$ of an algebraic number field of unit rank 1. It is shown how the infrastructure idea of Shanks can be used to decrease the number of binary operations needed to compute $R$ from the best known $O(RD^\varepsilon)$ for most continued fraction methods to $O(R^{1/2}D^\varepsilon)$. These ideas can also be applied to significantly decrease the number of operations needed to determine whether or not any fractional ideal of $\mathcal{O}$ is principal.

1. Introduction. In [16] Shanks introduced an idea which has since been modified and extended by Lenstra [13], Schoof [15] and Williams [17]. This idea can be used to decrease the number of binary operations needed to compute the regulator of a real quadratic order of discriminant $D$ from $O(D^{1/2+\varepsilon})$ to $O(D^{1/4+\varepsilon})$ for every $\varepsilon > 0$. In [21] and [17] Williams et al. showed that Shanks' idea could be extended to complex cubic fields. In this note we show that it can be further extended to any order $\mathcal{O}$ of an algebraic number field $\mathcal{F}$ of unit rank one, i.e., to orders of real quadratic, complex cubic, and totally complex quartic fields.

We present an algorithm which computes the regulator $R$ of $\mathcal{O}$ in $O(R^{1/2}D^\varepsilon)$ binary operations. Here, $D$ is the absolute value of the discriminant of $\mathcal{O}$. We also describe a method for testing an arbitrary (fractional) ideal $\mathfrak{a}$ of $\mathcal{O}$ for principality. This technique requires a number of binary operations that is $O(R^{1/2}D^\varepsilon + p(m))$, where $p(m)$ is a polynomial in the input length $m$ of $\mathfrak{a}$.

2. The Baby Step Algorithm. Let $\mathcal{F}$ have degree $n$ over the rationals $\mathbb{Q}$, and suppose $\mathcal{F}$ has $s$ real $\mathcal{Q}$-isomorphisms $\sigma_1, \sigma_2, \ldots, \sigma_s$ and $t$ pairs of complex $\mathcal{Q}$-isomorphisms $\sigma_{s+1}, \sigma_{s+1}, \ldots, \sigma_m, \sigma_m$ into $\mathbb{C}$, $m = s + t$. Since $\mathcal{F}$ has unit rank 1, we have $m = 2$, and we have only two normalized Archimedean valuations on $\mathcal{F}$, $| |_1$ and $| |_2$, where by $| \xi |_i$ we denote $| \sigma_i(\xi) | e_i$. Here, $e_i = 1$ when $\sigma_i$ is real and $e_i = 2$ when $\sigma_i$ is complex. As is usual in the unit theory, we introduce the logarithm mapping

$$\text{Log}: \mathcal{F}^\times \to \mathbb{R}$$

$$\xi \mapsto \text{Log } \xi = \log | \xi |_1.$$
Since \( m = 2 \), the image \( \mathcal{L} = \log(U) \) of the unit group \( U \) in \( \mathcal{O} \) is a one-dimensional lattice on the real line \( \mathcal{R} \). The regulator \( R \) is a basis of this lattice. It can be determined by

**Proposition 2.1.** Let \( \eta \) be a unit in \( \mathcal{O} \) such that \( \log \eta \) is the smallest positive value in \( \mathcal{L} \). Then \( \eta \) is a fundamental unit of \( \mathcal{O} \), and \( \log \eta \) is the regulator of \( \mathcal{O} \).

**Proof.** Since \( \mathcal{L} \) is a lattice of dimension one, each shortest nonzero vector in \( \mathcal{L} \) generates \( \mathcal{L} \). □

A first "naive" method for finding \( R \) is to walk in what we will call "baby steps" (cf. [11]) along the real line, starting at the origin \( O \) until we reach \( R \). We will now explain what is meant by these "baby steps".

Units \( \eta \) in \( \mathcal{O} \) have the property that there is no \( \alpha \neq 0 \) in \( \mathcal{O} \) such that \( |\alpha_i| < |\eta_i| \) for \( i = 1 \) and \( 2 \). This is true because \( |N(\alpha)| = |\alpha_1| |\alpha_2| (\alpha \in \mathcal{O}) \), \( N(\alpha) \in \mathcal{Z} \), and \( |N(\eta)| = 1 \). This property, however, does not completely characterize units, as there are many more elements of \( \mathcal{O} \) with this feature. Indeed, we now present

**Definition 2.2.** Let \( \mathfrak{a} \) be a (fractional) ideal of \( \mathcal{O} \). We call \( \mu \in \mathfrak{a} \) a minimum of \( \mathfrak{a} \) if there is no \( \alpha \neq 0 \) in \( \mathfrak{a} \) with \( |\alpha_i| < |\mu_i| \) for \( i = 1 \) and \( 2 \). The set of all minima of \( \mathfrak{a} \) is denoted by \( \mathfrak{M}_\mathfrak{a} \).

These minima have several important properties.

**Proposition 2.3.** Let \( \mathfrak{a} \) be a fractional ideal of \( \mathcal{O} \), let \( \xi \in \mathcal{T}_\mathfrak{a} \), and let \( \mu \in \mathfrak{M}_\mathfrak{a} \). Then \( \xi \mu \) is a minimum of \( \mathfrak{a} \). In particular, if \( \varepsilon \) is a unit of \( \mathcal{O} \), then \( \varepsilon \mu \) is a minimum of \( \mathfrak{a} \); that is, the unit group of \( \mathcal{O} \) acts on \( \mathfrak{M}_\mathfrak{a} \).

**Proof.** Clear. □

**Proposition 2.4.** Let \( \mathfrak{a} \) be a fractional ideal of \( \mathcal{O} \) and let \( \mu \in \mathfrak{M}_\mathfrak{a} \); then \( |N(\mu)| \leq \sqrt{D}N(\mathfrak{a}) \), where \( N(\cdot) \) denotes the norm.

**Proof.** As pointed out in Buchmann [5, Proposition 2.2], this statement is a consequence of Minkowski's convex body theorem. □

**Proposition 2.5.** Let \( \mathfrak{a} \) be a fractional ideal of \( \mathcal{O} \). Then \( \log \mathfrak{M}_\mathfrak{a} \) is a discrete set on the real line \( \mathcal{R}^{>0} \); and, for each point \( x \) on the real line, there are only finitely many minima \( \mu \in \mathfrak{M}_\mathfrak{a} \) with \( \log \mu \leq x \).

**Proof.** Select a constant \( c \in \mathcal{R}^{>0} \) and consider all the minima \( \mu \in \mathfrak{M}_\mathfrak{a} \) with \( |\log \mu| \leq c \). For any such minimum we have

\[
\exp(-c) \leq |\mu|^1 \leq \exp(c).
\]

But, from Proposition 2.4, we also know that

\[
|N(\mu)| = |\mu|^1 |\mu|^2 \leq \sqrt{D}N(\mathfrak{a}).
\]

Hence, from (2.1) and (2.2) we get

\[
|\mu|^1 \leq \exp(c) \quad \text{and} \quad |\mu|^2 \leq \sqrt{D}N(\mathfrak{a}) \exp(c).
\]

Since \( \mathfrak{a} \) is a free \( \mathcal{Z} \) module of rank \( n \), only a finite number of elements of \( \mathfrak{a} \) can satisfy (2.3). □

In particular, the set \( \Lambda = \log \mathfrak{M}_\mathfrak{a} \) is a discrete set in \( \mathcal{R} \) with subset \( \mathcal{L} \). Thus, we can write \( \Lambda \) as a sequence:

\[
\Lambda = (\lambda_j \in \mathcal{Z}).
\]
The ordering of the elements $\Lambda$ given in this sequence is uniquely determined by the condition
\[ \lambda_i < \lambda_j \iff i < j \quad \text{for } i, j \in \mathbb{Z}. \]

Taking a "baby step" means going from $\lambda_i$ to $\lambda_{i+1}$. Before we explain in further detail how this is done, we first note

**Proposition 2.6.** The sequence $(\lambda_i)$ is purely periodic modulo the regulator $R$ of $\mathcal{O}$.

**Proof.** Since the absolute norms of the minima of $\mathcal{O}$ are bounded by $\sqrt{D}$, there can only be finitely many pairwise, nonassociated minima. This means that $\Lambda$ modulo $R$ is finite. But, since by Proposition 2.3, $U$ acts on $M_{\mathbb{C}}$, the sequence must be purely periodic modulo $R$. \( \square \)

We remark here that Lenstra [13] and Schoof [15] in their description of the real quadratic case immediately consider the sequence $(\lambda_i) \mod R$. As, in this paper, it is $R$ which we wish to compute, we will approach this sequence in a somewhat different fashion.

We now describe the geometry of the sequence $(\lambda_i)$ somewhat further.

**Proposition 2.7.** (i) For every $k \in \mathbb{Z}$ we have $\lambda_{k+1} - \lambda_k < \log \sqrt{D}$.

(ii) For every $k \in \mathbb{Z}$ we have $\lambda_{k+j} - \lambda_k > c_1$ with

\[
j = \begin{cases} 
2, & n = 2, \\
7, & n = 3, \\
30, & n = 4,
\end{cases}
\]

and

\[
c_1 = \begin{cases} 
\log 2, & n = 2, \\
\log 4, & n = 3, \\
\log(2\cos(\pi/5)), & n = 4.
\end{cases}
\]

**Proof.** The proof of (i) is given in Buchmann [7]; the proof of (ii) can be found in [17] $n = 2$, Williams [18] $n = 3$, and Buchmann [4] $n = 4$. \( \square \)

**Corollary 2.8.** Let $p$ be the number of points in $(\lambda_i) \pmod{R}$; then

\[ R/(\log \sqrt{D}) < p < jR/c_1. \]

Corollary 2.8 shows that the number of baby steps necessary to compute $R$ is $O(R)$. In order to perform these baby steps we must first be able to answer

**Questions 2.9.** (i) How can one compute $\lambda_{k+1}$ from $\lambda_k$?

(ii) How can one decide whether or not $\lambda_k = R$?

To facilitate answering these questions, we introduce the mapping

\[ \phi: \mathcal{F}^\times \rightarrow \mathcal{P}(\mathcal{O}) \times \mathbb{R}, \]

\[ \alpha \mapsto \phi(\alpha) = (\phi_1(\alpha), \phi_2(\alpha)) = ((1/\alpha)\mathcal{O}, \log(\alpha)), \]

where by $\mathcal{P}(\mathcal{O})$ we denote the group of all nonzero principal ideals of $\mathcal{O}$. That is, we represent the elements in $\mathcal{F}^\times$ by a principal ideal of $\mathcal{O}$ and a real number. This representation has the following properties.
PROPOSITION 2.10. (i) $\phi$ is a group homomorphism whose kernel is the group of the roots of unity in $\mathcal{O}$.

(ii) The kernel of $\phi_1$ is the unit group $U$ of $\mathcal{O}$.

Proof. Since $\phi_1$ and $\phi_2$ are group homomorphisms, it follows that $\phi$ is a group homomorphism. Now if $\phi_1(\alpha) = \mathcal{O}$, then $1/\alpha \mathcal{O} = \mathcal{O}$, and we see that both $1/\alpha$ and $\alpha$ belong to $\mathcal{O}$. Hence $\alpha \in U$. If, moreover, $\phi_2(\alpha) = 0$, then $|\alpha|_2 = 1$; but, since $\alpha$ is a unit, this means that $|\alpha|_2 = 1$ and that $\alpha$ is a root of unity. \(\square\)

The last statement shows that $\phi$ represents each element of $\mathcal{F}^\times$ uniquely up to a root of unity. By looking at $\phi_1(\alpha)$, we can also tell whether $\alpha$ is a unit of $\mathcal{O}$; and, if $\alpha$ is a fundamental unit, then $|\phi_2(\alpha)|$ will be the regulator of $\mathcal{O}$.

When performing calculations, we represent the principal ideal $\phi_1(\alpha)$ by a $\mathbb{Z}$-basis. More precisely, we fix a $\mathbb{Z}$-basis $\omega_1, \omega_2, \ldots, \omega_n$ of $\mathcal{O}$. Then $\alpha = \phi_1(\alpha)$ is given by its denominator

$$d(\alpha) = \min\{d' \in \mathbb{Z}^+ \mid d' \alpha \subseteq \mathcal{O}\}$$

and an integral transformation matrix $A = (a_{ij}) \in \mathbb{Z}^{n \times n}$ with the property that the elements

$$\alpha_j = \left(\sum_{k=1}^{n} a_{jk} \omega_k\right)/d(\alpha) \quad (1 \leq j \leq n)$$

form a $\mathbb{Z}$-basis of $\alpha$. This matrix is uniquely determined up to a unimodular transformation from the left. We make the matrix $A$ unique by choosing it in some normal form, for example, Hermite normal form. In this case we write $A = \text{HNF}(\alpha)$, and we have $0 \leq a_{ij} < a_{jj}$ $(i < j)$, $a_{ij} = 0$ for $i > j$. Since $A$ and $d$ are unique for $\alpha$, we write $\alpha$ as $\phi(A, d)$. The advantage of this representation is that we can represent minima of $\mathcal{O}$ by small numbers.

PROPOSITION 2.11. Let $\mu$ be a minimum of $\mathcal{O}$, let $d = d(\phi_1(\mu))$, and let $A = \text{HNF}(\phi_1(\mu))$. Then

$$d \leq \sqrt{D} \quad \text{and} \quad |A|_\infty < \sqrt{D}.$$ 

Proof. The fractional ideal $\alpha = (1/\mu)\mathcal{O}$ contains 1 as a minimum. Hence the ideal $\alpha' = \alpha d$ is an integral primitive ideal which contains $d$ as a minimum. Moreover, $d$ must be the smallest positive integer contained in $\alpha'$, and we therefore find by Proposition 2.4 and the reasoning of Theorem 6.3 of [6] that

$$N(d) = d^n \leq N(\alpha') \sqrt{D} \leq d^{n-1} \sqrt{D},$$

which means that $d \leq \sqrt{D}$. Since $d \omega_j \in \alpha'$ and the numbers $\alpha_j$ $(j = 1, 2, \ldots, n)$ form a basis of $\alpha$, we have $a_{jj} \mid d$ $(j = 1, 2, 3, \ldots, n)$. \(\square\)

We remark that the order of magnitude of a minimum can be as large as $\exp \sqrt{D}$ (see, for example, Patterson and Williams [14]), which shows that the representation of a minimum $\mu$ by using $\phi$ is much better than the representation by means of the coefficients of the basis elements $\omega_1, \omega_2, \ldots, \omega_n$ of $\mathcal{O}$. Given this representation $\phi(\mu)$ of $\mu$, we are now able to answer Questions 2.9. We first prove

PROPOSITION 2.12. Let $k \in \mathbb{Z}$ and let $\mu_k \in M_\mathcal{O}$ with $\log \mu_k = \lambda_k$. Further, let $\eta$ be a minimum in $\phi_1(\mu_k)$ with minimal positive $\log \eta$. Then $\mu_{k+1} = \eta \mu_k \in M_\mathcal{O}$ and $\log(\mu_{k+1}) = \lambda_{k+1}$.

Proof. Follows easily from Proposition 2.3. \(\square\)
We are now able to present

**ALGORITHM 2.13 (The baby step method)**

*Initialization*

\[ A \leftarrow I_n \] (Identity matrix of order \( n \))

\[ d \leftarrow 1 \]

\[ R \leftarrow 0 \]

*Step 1 (Baby step)*

Compute in the ideal \( \alpha = \alpha(A, d) \) a minimum \( \eta \) with minimal positive \( \log \eta \). Set \( d \leftarrow d((1/\eta)\alpha) \), \( A \leftarrow \text{HNF}((1/\eta)\alpha) \), \( R \leftarrow R + \log \eta \).

*Step 2 (Ready ?)*

If \( d = 1 \) and \( A = I_n \), then \( R \) is the regulator of \( \mathcal{O} \) and the algorithm terminates; otherwise, go to Step 1.

The computation of \( \eta \) in Step 1 has been explained for \( n = 2 \) in [13], [15] and [17], for \( n = 3 \) in [17] and for \( n = 4 \) in Buchmann [5].

**PROPOSITION 2.14.** Algorithm 2.13 computes the regulator \( R \) of \( \mathcal{O} \) in \( O(RD^\varepsilon) \) binary operations on numbers of size \( O(D^\varepsilon) \).

*Proof.* By Corollary 2.8 the number of iterations in Algorithm 2.13 is \( O(R) \). By [17] and [5] it takes \( O(D^\varepsilon) \) binary operations to compute \( \eta \) in Step 1. Finally, by Proposition 2.11, the binary length of the numbers involved is \( O(D^\varepsilon) \). \( \square \)

3. **The Giant Step Algorithm.** Algorithm 2.13 is a very effective algorithm as long as \( D \) is small. It has been used, for example, by Williams and Broere [19] in the real quadratic case and by Angell [1] and Williams, Cormack and Seah [20] in the complex cubic case and Buchmann [5] in the totally complex quartic case. Other types of baby step algorithms have been used by Ince [11], Hendy [10] and Atkin (see Buell [3]). Unfortunately, as the values of \( D \) become very large, these methods become much too time-consuming. In fact, if \( \mathcal{O} \) is the maximal order of \( \mathcal{F} \) and if the class number of \( \mathcal{O} \) is small, then by the Brauer-Siegel Theorem [2] the regulator \( R \) of \( \mathcal{O} \) will be approximately of the same order of magnitude as \( \sqrt{D} \). By Corollary 2.8 this means that the number of iterations of Algorithm 2.13 will be approximately of the same order of magnitude as \( \sqrt{D} \). For example, in [14] it was found that for the maximal order of \( \mathcal{Q}(\sqrt{D}) \) with \( D = 350240722763374 \), the number of iterations is \( p = 70400728 \). Shanks [16] was the first to observe in the real quadratic case that it is possible to skip a large number of the baby steps by taking what we will call “giant steps”. In this section we will show that his idea applies to the unit rank 1 case in general.

Assume that we know the representations \( \phi(\mu_1) \) and \( \phi(\mu_2) \) of two minima \( \mu_1, \mu_2 \) in \( \mathcal{O} \), where, as before, \( \mathcal{O} \) is any order of \( \mathcal{F} \). Now we form \( \psi = \phi(\mu_1)\phi(\mu_2) \). Using a Hermite reduction, \( \psi \) can be computed in \( O(D^\varepsilon) \) binary operations (see Kannan and Bachem [12]). In general, \( \psi \) will not be the representation of a minimum of \( \mathcal{O} \); but, we can apply a certain reduction procedure to \( \psi = (a, \delta) \) in order to make it the representation of a minimum. For this purpose, we use one of the algorithms of [17] or Buchmann and Williams [8], [9] to obtain a minimum \( \eta \) in \( a \). Then we
put \( a^* = (1/\eta)a \) and \( \delta^* = \delta + \log \eta \) and we define the operation \( * \) by

\[
\phi(\mu_1)^* \phi(\mu_2) = (\phi_1(\mu_1)^* \phi_1(\mu_2), \phi_2(\mu_1)^* \phi_2(\mu_2)) = (a^*, \delta^*).
\]

**Proposition 3.1.** Let \( \mu_1, \mu_2 \in M_\mathcal{O} \).

(i) There is a minimum \( \mu^* \) in \( \mathcal{O} \) such that \( \phi(\mu_1)^* \phi(\mu_2) = \phi(\mu^*) \).

(ii) We have 

\[
-c_4 < \phi_2(\mu^*) - \phi_2(\mu_1 \mu_2) \leq c_5 \text{ with}
\]

\[
c_4 = \begin{cases} 
\log D, & \text{for } n = 2, \\
2 \log(D/3), & \text{for } n = 3, \\
\log 16D^5, & \text{for } n = 4,
\end{cases}
\]

and

\[
c_5 = \begin{cases} 
0, & \text{for } n = 2, \\
0, & \text{for } n = 3, \\
\log 16D, & \text{for } n = 4.
\end{cases}
\]

**Proof.** Since \( \eta \) is a minimum in \( a = (1/\mu_1 \mu_2) \mathcal{O} \), the element \( \mu^* = \eta \mu_1 \mu_2 \) must by Proposition 2.3 be a minimum in \( \mathcal{O} \), and \( \phi(\mu^*) = \phi(\mu_1)^* \phi(\mu_2) \). The bounds in (ii) for \( n = 2, 3 \) follow from estimates given in [17].

When \( n = 4 \), we note that \( \phi_1(\mu_1) \) and \( \phi_1(\mu_2) \) are reduced ideals; thus,

\[
d(a) = d(\phi_1(\mu_1) \phi_2(\mu_2)) \leq D.
\]

If we put \( d = d(a) \) and \( d' = da \), then \( N(d') \leq d^4 \). Thus, by using the algorithm of [8] we can find a minimum \( \mu' \) of \( d' \) such that

\[
|\mu'| \leq 4Wd.
\]

The latter inequality follows from (4.3) of [8]. By (3.1) we now have

\[
|\mu'|_i \leq 16W^2 d^2;
\]

but, since \( |N(\mu')| = |\mu'|_1 |\mu'|_2 \geq 1 \), we get

\[
|\mu'|_i \geq (16W^2 d^2)^{-1}.
\]

Now \( \eta = \mu'/d \) is a minimum in \( a \); hence,

\[
(16W^2 d^4)^{-1} \leq |\eta|_i \leq 16W^2.
\]

Since \( d < D, W < \sqrt{D} \) (see (2.2) of [8]), we find that

\[
(16D^5)^{-1} \leq |\eta|_i \leq 16D.
\]

Thus, we see that if we are given the representations \( \phi(\mu_1) \) and \( \phi(\mu_2) \) for two minima \( \mu_1, \mu_2 \) of \( \mathcal{O} \), we can make the giant step \( \phi(\mu_1)^* \phi(\mu_2) \); and, by Proposition 3.1(ii), we can almost precisely predict the value of \( \phi_2(\mu_1)^* \phi_2(\mu_2) \). This information is now used in

**Algorithm 3.2 (The giant step algorithm)**

**Initialization**

\[
\kappa \leftarrow 2c_4, K \leftarrow [\kappa]
\]

**Step 1 (Baby steps)**
By the method of Algorithm 2.13, compute the representations of the minima \( \mu \) in \( \mathcal{O} \) with

\[
\phi_2(\mu) \leq \kappa + c_5 + \log \sqrt{D}.
\]

If \( R \) is found, then terminate the algorithm. If not, store all these representations and sort them such that the denominators \( d \) and the HNF's, representing the first component of \( \phi(\mu) \), are in lexicographical order. We denote these representations by \( \phi^{(1)}, \phi^{(2)}, \ldots, \phi^{(j)} \), where \( \phi^{(i)} = (\phi_1^{(i)}, \phi_2^{(i)}) \).

**Step 2 (Choice of width of giant step)**

From \( \phi^{(1)}, \phi^{(2)}, \ldots, \phi^{(j)} \) choose \( \psi^* \) with

\[
\kappa \leq \psi^*_2 < \kappa + \log \sqrt{D}
\]

(This is possible by Proposition 2.7(i).) Set \( \Psi^{(0)} = \psi^*, i \leftarrow 0. \)

**Step 3 (Giant step)**

Compute

\[
\Psi^{(i+1)} = \Psi^{(i)} \ast \psi^*
\]

and put \( i \leftarrow i + 1. \)

**Step 4 (Test)**

If \( \Psi^{(i)} = \phi_1^{(k)} \) for some \( k \in \{1, 2, 3, \ldots, j\} \), then set \( R = \Psi^{(i)} - \phi_2^{(k)} \) and terminate the algorithm. (Of course, we determine whether or not \( \Psi^{(i)} = \phi_1^{(k)} \) by conducting a binary search of the first components of the baby stock \( \phi^{(1)}, \phi^{(2)}, \ldots, \phi^{(j)} \).)

**Step 5 (Increase \( \kappa \))**

If \( i = K \), then put \( \kappa \leftarrow 2\kappa, \ K = \lfloor k \rfloor \) and go to Step 1; otherwise, go to Step 3.

**Theorem 3.3.** Algorithm 3.2 computes the regulator \( R \) of \( \mathcal{O} \) in \( O(R^{1/2}D^\varepsilon) \) binary operations.

**Proof.** For a fixed \( \kappa \), and \( \psi^* \) fixed by Step 2, we have by Proposition 3.1 (ii)

\[
\Psi^{(j)}_2 = (j + 1)\psi^*_2 + \sum_{r=1}^{j} \varepsilon_r,
\]

where \(-c_4 < \varepsilon_r \leq c_5\) and \( j \leq K = \lfloor k \rfloor \). It follows that

\[
\Psi^{(k)}_2 > \kappa^2 - c_4\kappa.
\]

Thus, if

\[
\kappa > (R + c_4^2/4)^{1/2} + c_4/2,
\]

we must have \( \Psi^{(k)}_2 > R \). Thus, the first time we have a \( \kappa \) satisfying (3.3), we must have some \( i \ (1 \leq i \leq K) \) such that

\[
\Psi^{(i)}_2 > R \quad \text{and} \quad \Psi^{(i-1)}_2 < R.
\]

Since

\[
\Psi^{(i)}_2 = \Psi^{(i-1)}_2 + \psi^*_2 + \varepsilon_i,
\]

we must have \( \Psi^{(i)}_2 > (R + c_4^2/4)^{1/2} + c_4/2 \) and \( \Psi^{(i-1)}_2 < R \). Thus, the first time we have a \( \kappa \) satisfying (3.3), we must have some \( i \ (1 \leq i \leq K) \) such that

\[
\Psi^{(i)}_2 > R \quad \text{and} \quad \Psi^{(i-1)}_2 < R.
\]

Since

\[
\Psi^{(i)}_2 = \Psi^{(i-1)}_2 + \psi^*_2 + \varepsilon_i,
\]
we get
\[ R < \Psi_2^{(i)} < R + \kappa + \log \sqrt{D} + c_5. \]
It follows from (3.2), Proposition 2.6, and Proposition 2.10 that there must be some \( k \in \{1, 2, \ldots, j\} \) such that
\[ \Psi_2^{(i)} = \phi_1^{(k)} \quad \text{and} \quad R = \Psi_2^{(i)} - \phi_2^{(k)}. \]
For a fixed value of \( \kappa \), the number of binary operations performed by Step 1 of Algorithm 3.2 is, by the argument of the proof of Proposition 2.14, \( O(\kappa D^\epsilon) \). Also, the number of binary operations needed to compute a giant step is \( O(D^\epsilon) \); hence, for a fixed value of \( \kappa \) the entire algorithm performs \( O(\kappa D^\epsilon) \) binary operations. Since \( R = O(D^{1/4+\epsilon}) \), we know that we need to increase \( \kappa \) \( O(D^\epsilon) \) times until (3.3) first holds. It follows that in order to find \( R \), Algorithm 3.2 performs a total of \( O(R^{1/2} D^\epsilon) \) binary operations. □

4. Principal Ideal Testing. As already mentioned in [17] and [8], it is possible to modify the previous algorithm in order to produce a principal ideal test. To this end, we introduce the notion of a reduced ideal.

**Definition 4.1.** A (fractional) ideal \( a \) of \( \mathcal{O} \) is said to be reduced if 1 is a minimum in \( a \).

**Proposition 4.2.** Let \( a \) be any fractional ideal of \( \mathcal{O} \) and let \( \mu \) be a minimum in \( a \); then \((1/\mu)a\) is reduced.

**Proof.** Follows as a direct consequence of Proposition 2.3. □

Proposition 4.2 provides us with a method for computing a reduced ideal in the ideal class of any given ideal of \( \mathcal{O} \). Algorithms for doing this have been given in [17] and [8].

**Proposition 4.3.** Let \( a \) be a reduced ideal of \( \mathcal{O} \). Then \( a \) is principal if and only if there is a minimum \( \mu \) of \( \mathcal{O} \) with \( \phi_1(\mu) = a \) and \( 0 < \phi_2(\mu) < R \).

**Proof.** Buchmann [6, Theorem 6.2]. □

We are now able to present the following method for testing a given ideal \( a \) of \( \mathcal{O} \) for principality.

**Algorithm 4.4** (Principal ideal testing with baby steps)

**Step 1 (Computation of the reduced principal ideals)**
By the method of Algorithm 2.13 compute \( \phi_1(\mu) \) for every minimum \( \mu \) of \( \mathcal{O} \) with \( 0 < \phi_2(\mu) < R \). Store all these representations in terms of their denominators and their HNF’s and order them lexicographically.

**Step 2 (Reduction of \( a \))**
Compute a minimum \( \mu \) in \( a \) and put \( a^* = (1/\mu)a \). Store this ideal in terms of its denominator and HNF.

**Step 3 (Comparison)**
If \( a^* = \phi_1(\mu) \) for one of the representations computed in Step 1, then \( a \) is principal; otherwise, \( a \) is not principal.
As already proved in [8] we have

**Proposition 4.5.** Let \( a \) be a fractional ideal of \( \mathcal{O} \). Algorithm 4.4 tests \( a \) for principality in

\[
O(\log(|HNF(a)|_\infty d(a)) + D^\varepsilon \log(d(a)^n N(a)) + RD^\varepsilon)
\]

binary operations. \( \square \)

It is clear that Algorithm 4.4 will run very quickly when the number of reduced principal ideals of \( \mathcal{O} \) is small. If this is not the case, then we can again use the giant step technique to improve considerably the speed of this algorithm. We do this in

**Algorithm 4.6** (Principal ideal test with giant steps)

**Step 1** (Determination of the baby stock)

By the method of Algorithm 2.13 compute the representations of all the minima \( \mu \in \mathcal{O} \) with

\[
\phi_2(\mu) \leq \sqrt{R} + c_4 + c_5 + \log \sqrt{D}.
\]

Store all these representations in terms of their denominators and their HNF’s and order them lexicographically. Denote these representations by \( \phi^{(1)}, \phi^{(2)}, \ldots, \phi^{(j)} \), where \( \phi^{(i)} = (\phi^{(i)}_1, \phi^{(i)}_2) \).

**Step 2** (Reduction of \( a \))

Compute a minimum \( \mu \) in \( a \) and put \( a^* = (1/\beta)a \). Store this ideal in terms of its denominator and HNF.

**Step 3** (Initialize giant step procedure)

Put \( i \leftarrow 0, K = \lfloor \sqrt{R} \rfloor + 1 \). Find \( \phi^* \) in the baby stock such that

\[
\sqrt{R} + c_4 \leq \phi^*_2 < \sqrt{R} + c_4 + \log \sqrt{D}.
\]

Put \( \Psi_1^{(0)} = a^* \).

**Step 4** (Test)

If \( i > K \), then \( a \) is not a principal ideal and we terminate the algorithm. If \( i \leq K \) and \( \Psi_1^{(k)} = \phi_1^{(k)} \) for some \( k \in \{1, 2, 3, \ldots, j\} \), then \( a \) is principal and we terminate the algorithm.

**Step 5** (Giant Step)

Put

\[
\Psi_1^{(i+1)} = \Psi_1^{(i)} \ast \phi^*_1
\]

\[
i \leftarrow i + 1
\]

Go to Step 4.

**Theorem 4.7.** Let \( a \) be a fractional ideal of \( \mathcal{O} \). Algorithm 4.6 tests \( a \) for principality in

\[
O(\log(|HNF(a)|_\infty d(a)) + D^\varepsilon \log(d(a)^n N(a)) + R^{1/2}D^\varepsilon)
\]

binary operations.

**Proof.** Assume \( a \) is principal. By Propositions 4.2 and 4.3 there must be a minimum \( \mu \) of \( \mathcal{O} \) such that \( \phi_1(\mu) = a^* \) and \( 0 \leq \phi_2(\mu) < R \). We also have

\[
\Psi_2^{(k+1)} = k\phi^*_2 + \phi_2(\mu) + \sum_{r=1}^{k} \varepsilon_r,
\]

License or copyright restrictions may apply to redistribution; see https://www.ams.org/journal-terms-of-use
where \(-c_4 < \varepsilon_r \leq c_5\). If \(\phi_2(\mu) < \sqrt{R} + c_4 + c_5 + \log \sqrt{D}\), then Algorithm 4.6 will determine that \(a\) is principal when \(k = 0\). Otherwise,

\[
\Psi_2^{(k+1)} = \sum_{r=1}^{k} (\phi_2^* + \varepsilon_r) + \phi_2(\mu) > (k + 1)\sqrt{R}.
\]

Thus, when \(k = \lfloor \sqrt{R} \rfloor\) we have \(\Psi_2^{(k+1)} > R\). It follows that there must exist some \(i \ (1 \leq i \leq \lfloor \sqrt{R} \rfloor + 1 = K)\) such that

\[
\Psi_{2(i-1)} \leq R \quad \text{and} \quad \Psi_2^{(i)} > R.
\]

Since

\[
\Psi_2^{(i)} = \Psi_2^{(i-1)} + \phi_2^* + \varepsilon_i,
\]

we get

\[
R < \Psi_2^{(i)} < R + \sqrt{R} + c_4 + \log \sqrt{D} + c_5.
\]

It follows by Propositions 2.6 and 2.10 that

\[
\Psi_1^{(i)} = \phi_1^{(k)} \quad \text{for some } k \in \{1, 2, 3, \ldots, j\}.
\]

On the other hand, if \(a\) is not principal, then \(\Psi_1^{(i)} \sim a\) cannot be principal; thus, \(\Psi_1^{(i)} \neq \phi_i^{(k)}\) for any \(i\) or \(k\).

By the same arguments as those used in the proofs of Proposition 4.5 and Theorem 3.3 we see that Algorithm 4.6 will execute in

\[
O(\log(|\text{HNF}(a)|) d(a) + D^8 \log(d(a)^n N(a)) + R^{1/2} D^8)
\]

binary operations. $$\square$$

Mathematisches Institut
Universität Düsseldorf
4000 Düsseldorf 1, West Germany

Department of Computer Science
University of Manitoba
Winnipeg, Manitoba, Canada R3T 2N2