

# An accelerated Buchmann algorithm for regulator computation in real quadratic fields

Ulrich Vollmer\*

Technische Universität Darmstadt, Fachbereich Informatik  
Fachgebiet Kryptographie und Computeralgebra  
Alexanderstr. 10, 64283 Darmstadt

**Abstract.** We present a probabilistic algorithm for computing the regulator  $R$  of a real quadratic order of discriminant  $\Delta$  running in time  $L(\frac{1}{2}, 3/\sqrt{8} + o(1))$ .

## 1 Introduction

In his paper [Buc90], Buchmann proposed a generalization of Hafner and McCurley’s subexponential algorithm for class group computation in imaginary quadratic fields [HM89] to the computation of class group and regulator of arbitrary number fields. While his algorithm depends on an as yet unproven “smoothness assumption for reduced ideals” for fields of degree exceeding two, it does extend unconditionally Hafner and McCurley’s algorithm to real quadratic fields.

In this paper we present two modifications of Buchmann’s algorithm for the real quadratic case. Their goal is to improve the asymptotics of the expected run time. Correctness, and running time bounds for both algorithms depend on a Generalized Riemann Hypothesis (GRH).

The expected run time needed by Buchmann’s original algorithm in order to compute class group and regulator of a number field with discriminant  $\Delta$  and fixed degree was bounded by  $L_{|\Delta|}(\frac{1}{2}, 1.7)$  where

$$L_{|\Delta|}(a, b) = \exp(b(\log \Delta)^a (\log \log \Delta)^{1-a}).$$

Our first algorithm, RQCLR, computes class group and regulator of a real quadratic order with discriminant  $\Delta$  in time  $L_{\Delta}(\frac{1}{2}, \sqrt{2})$ . It confirms the correctness of its result by computing an approximation to the special value of the L-function of the field at 1.

The second algorithm, RQR, computes only the regulator in time  $L_{\Delta}(\frac{1}{2}, 3/\sqrt{8})$ . It produces with probability given a priori the correct result. However, it does not verify the correctness of the result.

The results of this paper are collected in the following theorem.

**Theorem 1.** (GRH) *For any positive real number  $p \leq 1$ , and  $\epsilon > 0$ , there is some  $M = M(\epsilon)$ , and a probabilistic algorithm that has the following property:*

---

\* research supported by the DFG

Given the positive discriminant  $\Delta > M$  of the quadratic order  $\mathcal{O}$ , the algorithm computes an integer  $R$  that differs from some positive multiple  $m \cdot R_\Delta$  of the regulator  $R_\Delta$  of  $\mathcal{O}$  by less than one. Independent of  $\Delta$ , the probability that  $m = 1$  taken over all random input of the algorithm is at least  $p$ .

The expected run time of the algorithm is bound by  $L_\Delta(\frac{1}{2}, c)$  where

a.  $c = 3/\sqrt{8} + \epsilon$  if  $p < 1$ ;

b.  $c = \sqrt{2} + \epsilon$  if  $p = 1$ .

In case b, the algorithm also computes the class number, and the elementary divisors of the class group of  $\mathcal{O}$ .

## 2 Previous work

The details of Buchmann's algorithm for the quadratic case were spelled out in Abel's thesis [Abe94]. Her algorithm is applicable to arbitrary quadratic orders, not only maximal ones. Abel was able to prove on the basis of some Generalized Riemann Hypothesis (GRH) that her variant of the algorithm runs in time bound by  $L_\Delta(\frac{1}{2}, 5/6\sqrt{3} + o(1))$ .

In [Vol00], Vollmer indicated briefly that three subalgorithms used by Abel can be substituted by faster ones. Mentioned were:

- Replacement of the factorisation algorithm used in the process of generating relations. This suggestion was already made in [HM89].
- Computation of an approximation of the regulator from logarithms of units that form a generating set of the unit group with the help of an algorithm proposed by Maurer in his thesis [Mau00];
- Use of the fast algorithm for computation of the determinant of the relation lattice proposed in [Vol00] itself.

This paper takes up the suggestions of [Vol00], incorporating them into a complete algorithm.

**Practical implementation.** The focus of this paper is in presenting an algorithm whose complexity can be rigorously proved (assuming GRH), although some of the ideas might also lead to practical improvements.

For advice on the practical implementation of Buchmann's algorithm for quadratic fields, we refer the reader to [Coh93], and [Jac99]. [Coh93] gives a detailed description of the algorithm as implemented in the well-known PARI package. (Please refer to the fourth printing, and the author's web site for the corrected text of the relevant passage.)

[Jac99] shows how the Multiple Quadratic Polynomial Sieve can be employed for rapid generation of relations in the quadratic case. The resulting algorithm is implemented in the LiDIA package of Buchmann et al.

To the best of our knowledge there are no published rigorous, or heuristic analyses of the expected run times of the algorithms proposed in the cited works. It is, however, to be expected that they share the asymptotic behaviour of RQR, or RQCLR, depending on the linear algebra algorithms employed.

### 3 Overview

Let  $\mathcal{O}$  be a real quadratic order, and  $K$  its fractional field which we assume to be embedded into  $\mathbb{R}$ . For simplicity we will assume in our exposition that  $\mathcal{O}$  is maximal. The restriction to this case instead of that of an arbitrary order is not essential. Minor modifications lead to an algorithm for the general case.

We denote the discriminant of  $\mathcal{O}$  by  $\Delta$ , the group of invertible  $\mathcal{O}$ -ideals by  $\mathcal{I}_\Delta$ , its subgroup of principle ideals by  $\mathcal{P}_\Delta$ , the class group  $\mathcal{I}_\Delta/\mathcal{P}_\Delta$  of  $\mathcal{O}$  by  $Cl_\Delta$ , the class number by  $h_\Delta$ , the regulator by  $\mathcal{R}_\Delta$ , and the non-trivial automorphism of  $K$  by  $\sigma$ .

We assume in the following that  $\mathcal{R}_\Delta \gg \log \Delta$ , since otherwise there are deterministic algorithms that are more efficient than the probabilistic ones proposed here.

Buchmann's algorithm uses the fact that we can compute in each ideal class "small" representatives, called reduced ideals, in polynomial time. For background on reduced ideals, the properties of the reduction operator, and cycles of reduced ideals we refer the reader to [Len82]. Here, we will just give the definition.

**Definition 1.** *An integral ideal  $\mathfrak{a} \in \mathcal{I}_\Delta$  is called primitive if  $\mathfrak{a} \subseteq q\mathbb{Z}$  implies  $q = 1$ . It is called reduced if it is primitive, and  $q = \min(\mathfrak{a} \cap \mathbb{N})$  is a minimum of  $\mathfrak{a}$ , i.e.  $|\alpha|, |\alpha^\sigma| < q$  imply  $\alpha = 0$  for any  $\alpha \in \mathfrak{a}$ .*

This definition coincides with the classical one introduced by Gauss in the language of binary forms.

In [Buc90], Buchmann introduced, generalising ideas by Seysen [Sey87], and Hafner/McCurley [HM89], lattices  $L = L^{(m)} \in \mathbb{Z}^m \oplus \mathbb{R}$  with determinant  $hR$ , and showed how to produce a generating set for  $L^{(m)}$  for suitably chosen  $m \gg 0$ .

We recall the definition of  $L^{(m)}$ . Roughly spoken, it is the lattice of "relations" over a large set of prime ideals.

We define the relevant set of prime ideals of  $\mathcal{O}$ . For  $b \in \mathbb{R}$ , let  $\mathcal{F}_b = \{\mathfrak{p} \in \mathcal{I} \mid N\mathfrak{p} < b \text{ prime}\}$ . Set  $c = L(\frac{1}{2}, z)$ , where  $z$  is later chosen such that  $\mathcal{F}_c$  is large enough for random reduced ideals to factor over  $\mathcal{F}$  with sufficiently high probability. The cardinality of  $\mathcal{F}$  will be denoted by  $m$ .

Let  $\mathcal{I}_c$  denote the free subgroup of  $\mathcal{I}_\Delta$  generated by  $\mathcal{F}$ . This is identified with  $\mathbb{Z}^m$  in the natural way. The algorithms presuppose that the restriction of the projection  $\psi : \mathcal{I}_\Delta \rightarrow Cl_\Delta$  to  $\mathcal{I}_c$  is surjective. Due to a well known result of Bach, cf. [Bac90] this is certainly the case if  $c > k = 6 \log^2 \Delta$  which we will henceforth assume throughout. Denote  $\mathcal{F}_k$  by  $\mathcal{G}$ , and  $\text{card } \mathcal{G}$  by  $l$ .

Let

$$\begin{aligned} \phi : K^* &\longrightarrow \mathcal{P}_\Delta : \alpha \longmapsto (\alpha), \\ \text{Log} : K^* &\longrightarrow \mathbb{R} : \alpha \longmapsto \frac{1}{2} \log \left| \frac{\alpha}{\alpha^\sigma} \right|, \end{aligned}$$

and  $\mathcal{O}_c = \phi^{-1}(\mathcal{I}_c \cap \mathcal{P}_\Delta)$ .

We define the lattice  $L^{(m)}$  to be the image of  $\mathcal{O}_c$  under  $(\text{Log}, \phi)$ . We will call its elements *relations*. The pre-image of a relation under  $(\phi, \text{Log})$  is called its generator. From the diagram

$$\begin{array}{ccccccc}
 1 & \longrightarrow & \pm 1 & \longrightarrow & \mathcal{O}_c & \xrightarrow{(\phi, \text{Log})} & \mathbb{Z}^m \oplus \mathbb{R} \\
 & & \downarrow & & \parallel & & \pi \downarrow \\
 1 & \longrightarrow & \mathcal{O}^* & \longrightarrow & \mathcal{O}_c & \xrightarrow{\phi} & \mathbb{Z}^m & \xrightarrow{\psi} & Cl_\Delta & \longrightarrow & 1
 \end{array}$$

we see that  $\pi|_{L^{(m)}}$  has kernel  $(0, \mathcal{R}_\Delta \mathbb{Z})$  and the sequence

$$0 \longrightarrow \mathbb{R}/\mathcal{R}_\Delta \mathbb{Z} \longrightarrow (\mathbb{Z}^m \oplus \mathbb{R})/L^{(m)} \longrightarrow Cl_\Delta \longrightarrow 1$$

is exact.

For any  $v = (\mathbf{v}, \text{Log } \alpha) \in L^{(m)}$ , we call  $\mathbf{v} = \pi(v)$  its integral part. For any sublattice  $M \subseteq L$ , we will denote  $\pi(M)$  also simply by  $M'$ .

Both RQR, and RQCLR compute  $\mathcal{R}_\Delta$  by producing couples of elements of  $L^{(m)}$  with the same image under  $\pi$ . To achieve this they proceed roughly in the following manner:

1. Construct the elements of the factor base  $\mathcal{F}$ .
2. Choose some  $n \in \mathbb{N}$ . For each  $j$  with  $1 \leq j \leq n$  generate a random relation  $v_j \in L^{(m)}$  and enter its coefficients into a matrix  $A$ . (Instead of the value of  $\text{Log}$ , we record its argument in compact representation.)
3. RQCLR *only*: Compute the determinant  $\tilde{h}$  of the column space of  $A$ , and its Hermite Normal Form (HNF)  $H$ .
4. Choose randomly two relations  $v_i$  with generators  $\alpha_i$ ,  $i = 1, 2$ . Express each  $\pi(v_i)$  as a linear combination of  $\pi(v_j)$  with  $j \neq i$ . Each found expression yields an element  $E_i$  of the kernel of  $\pi$ .
5. Compute the real GCD  $\tilde{R}$  of  $E_1$  and  $E_2$ .
6. RQCLR *only*: Calculate bounds for the product of class number and regulator using the L function of field  $K$ . If  $\tilde{h}\tilde{R}$  does not lie within these bounds, start over.
7. Output  $R' = \tilde{R}$ , and, if we are in RQCLR, also  $h = \tilde{h}$ .
8. RQCLR *only*: Compute the Smith Normal Form of  $H$ , and extract the class group structure.

The algorithms differ in the relation generation in step 2. In RQR we choose  $n$  large and compute many sparse relations, in RQCLR fewer, but mostly dense ones. The reason for the different asymptotic behaviour of RQR, and RQCLR lies in this difference. *NB*: in practical implementations one chooses  $n$  only slightly larger than  $m$ , and generates only sparse relations. It is still unclear why this succeeds.

We outline the rest of the paper. In the sections 4 through 6 we will treat those aspects of the proposed algorithms that are specific to our approach: the generation of random reduced ideals in an ideal class; and the extraction of a generating set of units from the relation matrix.

In section 7 we will list results from previous work, give the remaining details of RQR and RQCLR, and conclude the proof of Theorem 1.

## 4 Random relations

In [Buc89], Buchmann has given, and analysed a method for the construction of a generating system for the lattice  $L^{(m)}$  in the case of an arbitrary number field. This method relies in the real-quadratic case on the following proposition which can be proved in analogy to Propostion 4.4 of [Sey87] giving the same result for the imaginary quadratic case.

**Proposition 1.** (GRH) *The number  $N_c$  of reduced  $\mathcal{O}$ -ideals that factor completely over the ideals with prime norm smaller than  $c = L(\frac{1}{2}, z)$  is at least  $hR \cdot L(\frac{1}{2}, -1/(4z))$ .*

Buchmann proceeds by taking power products over  $\mathcal{F}$  with exponents up to  $\Delta$ , and choosing—by a method called PV—a random reduced ideal in the resulting class. For ease of reference, we will describe a simple variant of PV for the real-quadratic case which we will call RANDOMREDUCED that enjoys—with minor modifications—the same properties as the more general algorithm.

Another, slightly more elaborate variant of PV was given by Abel in her thesis [Abe94].

Let  $\mathfrak{a} \in I$  be some invertible  $\mathcal{O}$ -ideal. For any  $d \in \mathbb{N}$  we define the set

$$S_d = S_d(\mathfrak{a}) = \{(\mathfrak{b}, \alpha) \mid \mathfrak{b} \text{ is reduced, } \mathfrak{b} = \alpha\mathfrak{a}, d \leq \text{Log } \alpha / \log \Delta < (d+1)\}.$$

Let  $M > R$  be given. RANDOMREDUCED proceeds as follows.

1. Choose some random  $d \in [0, M)$ .
2. Enumerate all elements in  $S_d$ .
3. Choose randomly among them.

The following lemmata are needed to show that RANDOMREDUCED has the desired properties. For any  $M$  we denote by  $T_M$  the range of values of RANDOMREDUCED

$$T_M = T_M(\mathfrak{a}) = \bigcup_{d=0}^M S_d(\mathfrak{a}_i).$$

**Lemma 1.** *Fix  $\mathfrak{a} \in I$ . Let  $d \geq 1$ . Then  $2 \leq \text{card } S_d(\mathfrak{a}) \leq 2 \log_2 \Delta$ , and  $2M \leq \text{card } T_M(\mathfrak{a}) \leq 2M \log_2 \Delta$ .*

This is a trivial consequence of the properties of the reduction operator  $\rho$  proved in [Len82].

**Lemma 2.** *Given  $\mathfrak{a}, \mathfrak{b} \in I$ , where  $\mathfrak{a} \sim \mathfrak{b}$ , and  $\mathfrak{b}$  is reduced. Let  $d \geq 1$ . Then  $\text{card}\{d \mid \mathfrak{b} \in S_d(\mathfrak{a})\} = M \log \Delta / R + e$  with  $-2 \leq e \leq 1$ .*

*Proof.* Let  $\mathfrak{b} = \alpha\mathfrak{a}$ , where  $\alpha$  is chosen such that  $0 \leq \text{Log } \alpha < R$ . Then  $\mathfrak{b} \in S_d$  if and only if  $d \log \Delta \leq \text{Log } \alpha + kR < (d+1) \log \Delta$  for some  $k \in \mathbb{Z}$ . Since we assumed that  $R \gg \log \Delta$  the claim follows.

Let  $M \leq \Delta$ , and  $d \in [1, M]$ . We show that it is possible to enumerate all elements in  $S_d$  in polynomial time. For this to be possible, the field elements need to be given in compact representation. The following lemma follows immediately from results in [BTW95].

**Lemma 3.** *Given  $\alpha \in K$  in compact representation,  $\mathfrak{a} \in I$ , and  $n \in \mathbb{N}$ , it is possible to compute the compact representation of  $\alpha^n$ , and  $\alpha \mathfrak{a}$  in time polynomial in the size of  $\alpha$ , and  $\log n$ .*

Thus we may proceed as follows.

1. compute some  $\alpha$  with  $\text{Log } \alpha \in [\frac{1}{2} \log \Delta, \log \Delta]$ ;
2. compute  $l = \text{Log } \alpha$  with precision  $\log_2 \Delta$ ;
3. set  $k = \lfloor d \cdot (\log \Delta / l) \rfloor$ , and compute  $\alpha_1 = \alpha^k$ ;
4. compute  $\beta_0$  such that  $\mathfrak{b} = \beta_0 \alpha_1 \mathfrak{a} = \rho_0(\alpha_1 \mathfrak{a})$  is reduced;
5. compute  $\text{Log}(\beta_0 \alpha_1)$ , and—through successive reduction— all  $\beta_i$  such that  $\beta_i \mathfrak{b}$  is reduced, and  $\text{Log } \beta_i + \text{Log}(\beta_0 \alpha_1) \in [d \log \Delta, (d + 1) \log \Delta]$ .

Note that we can assure that all reduced ideals in  $S_d$  get enumerated, but due to the imprecise computation of logarithms in this enumeration process, the enumeration may inadvertently contain ideals with relative generators from a slightly larger interval. Since at most 2 ideals are thus erroneously listed, this will not affect the probability estimates that follow, and is, hence, ignored.

Note further that the ideal  $\mathfrak{a}$  might already be given as the product of a principle ideal (with generator  $\gamma$  in compact representation) with a reduced ideal  $\mathfrak{c}$ . In this case we start from  $\mathfrak{c}$ , and adjust  $k$  in step 3 accordingly.

We summarise the properties of RANDOMREDUCED in the following proposition.

**Proposition 2.** *Let  $\mathfrak{a}$  be a given invertible  $\mathcal{O}$ -ideal. RANDOMREDUCED computes randomly in polynomial time some  $\mathfrak{b} \in I$ , and  $\alpha \in K$  in compact representation such that  $\mathfrak{b} = \alpha \cdot \mathfrak{a}$  is reduced.*

*For any reduced  $\mathfrak{b}$  equivalent to  $\mathfrak{a}$  the probability that RANDOMREDUCED outputs  $\mathfrak{b}$  on input  $\mathfrak{a}$  is contained in the interval  $(\log(2)/R - 1/M, \log \Delta / (2R) + 1/M)$ . Moreover, the probability that the second component  $\alpha$  of the output of RANDOMREDUCED fulfills  $\text{Log } \alpha \in [kR, (k + 1)R)$  conditional on the fact that the first component is some fixed  $\mathfrak{b}$  is bounded from below by  $1/N$  with  $N = \lfloor M \log \Delta / R \rfloor - 1$  if  $k < (M \log \Delta - R) / R$ .*

*Proof.* All but the last claim follow in a straightforward manner from the preceding lemmata. We turn to the latter.

Fix some reduced  $\mathfrak{b}$  in the ideal class of  $\mathfrak{a}$ . Let  $\alpha_0$  be a generator of  $\mathfrak{b}$  relative to  $\mathfrak{a}$  with  $0 \leq \text{Log } \alpha_0 < R$ . Let further  $B = \{d \mid \exists m \text{ such that } d \leq (\text{Log } \alpha_0 + mR) / \log \Delta < d + 1\}$ . Then the sought conditional probability is certainly bounded from below by  $1/N$  where  $N = \text{card } B$ .

Now,  $0 \leq d < M$ , and  $d \leq (\text{Log } \alpha_0 + mR) / \log \Delta < d + 1$  imply  $0 \leq m < M \log \Delta / R$ . The claim follows.

We are now in the position to show how to generate random relations. The procedure will be called `RANDOMRELATION`.

Fix some  $\mathcal{H}$  with  $\mathcal{G} \subseteq \mathcal{H} \subseteq \mathcal{F}$  that parametrizes `RANDOMRELATION` in the sense that it determines whether we generate sparse ( $\mathcal{H} = \mathcal{G}$ ), or dense ( $\mathcal{H} = \mathcal{F}$ ) relations. Let  $\mathfrak{q} \in \mathcal{I}_\Delta$  be some ideal which will later be chosen to be some power of an element of  $\mathcal{F}$  that “offsets” the relation at one place.

1. For each  $\mathfrak{p} \in \mathcal{H}$  choose  $a_{\mathfrak{p}}$  with  $|a_{\mathfrak{p}}| \leq \Delta$ . Set  $a_{\mathfrak{p}} = 0$  for  $\mathfrak{p} \in \mathcal{F} \setminus \mathcal{H}$ .
2. Compute  $\mathfrak{a} = \mathfrak{q} \cdot \prod_{\mathfrak{p} \in \mathcal{H}} \mathfrak{p}^{a_{\mathfrak{p}}}$ .
3. Compute  $(\mathfrak{b}, \alpha) = \text{RANDOMREDUCED}(\mathfrak{a})$  with  $M = \Delta$ .
4. **if**  $\mathfrak{b} \notin \mathcal{I}_c$  **then return FAILURE**.
5. Compute  $b_{\mathfrak{p}}$  such that  $\mathfrak{b} = \prod_{\mathfrak{p} \in \mathcal{F}} \mathfrak{p}^{b_{\mathfrak{p}}}$
6. **return**  $((a_{\mathfrak{p}} - b_{\mathfrak{p}})_{\mathfrak{p} \in \mathcal{F}}, \alpha)$ .

In step 2, each computation of an ideal product is followed by reduction. Hence, the ideal  $\mathfrak{a}$  computed in step 2 is computed and stored as the product of some  $\alpha_0 \in K^*$  (in compact representation) and a reduced ideal.

For steps 4, and 5 we factor the norm of  $\mathfrak{b}$  with the elliptic curve method, cf. Algorithm 7.2 of [LP92].

**Lemma 4.** *For any class  $C \in Cl_\Delta$ , the probability that  $\mathfrak{a}$  computed in step 2 belongs to  $C$  is contained in an interval  $((1 - o_1(\Delta))/h, (1 + o_1(\Delta))/h)$  with  $o_1(\Delta) = o(1)$ .*

*Proof.* This lemma follows from lemma 4.5 of [Sey87].

**Lemma 5.** *For the probability  $p$  that a given reduced ideal is computed in step 3 we have  $hR \cdot p \in (\log(2) - o_2, \log \Delta + o_2)$  for some  $o_2 = o(1)$ .*

*Proof.* This follows from Proposition 2, and Lemma 4.

**Corollary 1.** *The probability that the ideal  $\mathfrak{b}$  computed in step 3 lies in  $\mathcal{I}_c$  is bounded from below by  $(\log(2) - o(1))L(\frac{1}{2}, -1/(4z))$ .*

*Proof.* Consequence of Proposition 1, and 2, and Lemma 5.

The repeated call to the procedure above with identical parameters until it returns successfully yielding some relation  $(\mathfrak{c}, \alpha)$  will be called `RANDOMRELATION`.

## 5 Relation lattices

In this section we estimate the number of relations which need to be generated to achieve one of the following two goals:

1. the lattice generated by the integer parts of the obtained relations equals  $L'$ ;
2. the likelihood that the integer part of a randomly chosen relation is contained in the lattice generated by the integer parts of the other relations exceeds some a priori given bound.

Moreover, we give a bound for the number of calls to RANDOMRELATION needed to obtain the necessary number of relations.

Both algorithms, RQR and RQCLR, start out by generating  $m$  relations whose integral parts form a square diagonally dominant matrix, as originally proposed by Seysen.

1. **for**  $i = 1$  **to**  $m$
2.  $(\mathbf{v}_i, \alpha_i) \leftarrow \text{RANDOMRELATION}(\mathcal{G}, \mathfrak{p}_i^{2^m \Delta})$

Let  $A_0$  denote the matrix containing the integral parts  $\mathbf{v}_i$  of the relations  $v_i$  generated this way, and  $L_0$  the lattice generated by  $\{v_i\}$ . Then  $\log_2 \det A_0 = \log_2 [L' : L'_0] < m \log_2 \Delta (1 + o_3)$  where  $o_3 = o(1)$  can be explicitly given.

**Lemma 6.** *Let  $(v_i), i = 1, \dots, n$  be a sequence of relations  $v_i \in L$ . Let further for any  $j = 1, \dots, n$  the sublattice  $L_j \subseteq L$  be generated by  $L_0$ , and all  $\pi(v_i) = \mathbf{v}_i$  with  $i \leq j$ . Then we have  $\mathbf{v}_{j+1} \in L'_j$  for at least  $n - m \log_2 \Delta (1 + o_3)$  values of  $j$ .*

*Proof.* This follows from the fact that any chain of sublattices  $M_i \subset L$  with  $L_0 \subset M_1 \subset \dots \subset M_e \subset L$  has length  $e$  smaller than  $(1 + o_3)m \log_2 \Delta$ .

Thus we only need to produce  $n = (1 + o_3)m \log_2 \Delta / (1 - p)$  additional relations  $v_i$  with RANDOMRELATION in order to ensure that with probability  $p$  a relation randomly chosen from among them is contained in the lattice generated by the rest.

**Lemma 7.** *Given some  $\mathbf{v} = (v_i) \in L'$  with  $0 \leq v_i \leq \Delta - \log \Delta$ , the probability that a call to RANDOMRELATION( $\mathcal{F}, (1)$ ) yields a  $v$  with  $\pi(v) = \mathbf{v}$  is at least  $(1 - o(1))h / (2\Delta^m \log_2 \Delta)$ .*

*Proof.* Let  $\mathfrak{c}$  correspond to  $\mathbf{v}$ , and let  $\mathfrak{b}$  run through the set of all  $c$ -smooth reduced ideals. RANDOMRELATION arrives at some  $v$  with  $\pi(v) = \mathbf{v}$  if it chooses in step 2 the ideal  $\mathfrak{c} \cdot \mathfrak{b}$ , which has exponents smaller than  $\Delta$  at each place by assumption, and  $\mathfrak{b}$  in step 3. For the second choice there are  $N_c$  possibilities differing in probability by a factor of  $2 \log_2 \Delta$ . Each such choice can follow  $(1 + o_1)D^m/h$  different power products every one of which occurs with the same probability. The claim follows.

Next we prove an estimate for the number of lattice points of  $L$  that are not in some sublattice of full rank. Define  $B(d) = \{(v_i) \in \mathbb{Z}^m \mid 0 \leq v_i \leq d\}$ .

**Lemma 8.** *Let  $M'$  be some proper sublattice of  $L'$ . Then  $L' \setminus M'$  contains at least  $(\Delta - 2h)^m / (2h)$  elements in  $B(\Delta)$ .*

*Proof.* We know that there is a basis of  $L'$  with positive coefficients smaller than  $h$ . Let  $\mathbf{w}$  be an element of that basis that is not in  $M'$ . Then we can assign to each  $\mathbf{v} \in M' \cap B(\Delta - h)$  the lattice point  $\mathbf{v} + \mathbf{w} \in B(\Delta)$  which is obviously in  $L' \setminus M'$ .

Now  $L \cap B(\Delta - h)$  contains at least  $(\Delta - 2h)^m/h$  elements. Thus we have either  $\text{card}((L' \setminus M') \cap B(\Delta - h)) \geq (\Delta - 2h)^m/2h$ , in which case we are done, or  $\text{card}(M' \cap B(\Delta - h)) \geq (D - 2h)^m/2h$ . Using the assignment from the previous paragraph we find again the desired number of elements in  $(L' \setminus M') \cap B(\Delta)$

Combining the last two lemmata we obtain an estimate for the probability that a call to `RANDOMRELATION` enlarges the relation lattice.

**Proposition 3.** *Let  $L'_k \subseteq L'$  be the lattice generated by  $L'_0$  and  $\mathbf{c}_i$  where  $(\mathbf{c}_i, \alpha_i)_{i=1}^n$  are obtained by calls to `RANDOMRELATION`( $\mathcal{F}, (1)$ ). Assume  $\text{rank } L'_k = m$ . Then the probability that the next call to `RANDOMRELATION` results in a vector  $\mathbf{c} \in L' - L'_k$  provided that the latter is nonempty, is bounded from below by  $(1 - o(1))/(4 \log \Delta)$ .*

If  $L'_n = L'$  then we call the corresponding  $m \times n$  matrix  $A$  a *full relation matrix*. The last proposition yields finally the desired conclusion about the number of relations we need to compute in order to arrive at a full relation matrix.

**Corollary 2.** *There is an effectively computable function  $o_2 = o_2(\Delta) = o(1)$  such that for  $n = L(\frac{1}{2}, z + o_2)$  the probability that  $L'_n = L'$  is bounded from below by  $1/2$ .*

## 6 Extracting a Generating Set of Units

In this section we assume that we are given the following data:

- Some  $m \times n$  relation matrix  $A = (a_{ij})$  with vector of generators  $\alpha_j$ . We have  $\prod_i \mathfrak{p}^{a_{ij}} = (\alpha_j)$ .
- Two sparse relations  $(\mathbf{v}_s, \gamma_s)$ ,  $s = 1, 2$ , obtained through a call to `RANDOMRELATION`( $\mathcal{G}, (1)$ ).
- Two vectors  $\mathbf{w}_s$  with  $A\mathbf{w}_s = \mathbf{v}_s$ .

We have seen in Lemma 6 how to find a  $\mathbf{v}$  which lies in the column space of a sparse relation matrix. If, on the other hand, we choose to compute dense relations, then Corollary 2 assures us that we can quickly compute a full relation matrix. Two more calls to `RANDOMRELATION` yield the desired dependent relations.

The vectors  $\mathbf{w}_s$  are computed with the algorithm `DIOPHANTINE SOLVER` proposed in [MS99]. This algorithm finds a solution to the Diophantine system  $A\mathbf{x} = \mathbf{v}$  with size restricted by

$$(1) \quad \log\|\mathbf{x}\| = O(m \log(m\|A\|)) + \log\|\mathbf{v}\|$$

On the basis of the above data, we can assign a unique unit to each relation vector:  $\epsilon_s = \gamma_s / \prod \alpha_j^{w_j}$  is a unit of  $\mathcal{O}$ , since  $\gamma_s$ , and  $\prod \alpha_j^{w_j}$  generate the same integral ideal. We denote  $\epsilon_s$  by `UNIT`( $\gamma_s, A, \mathbf{w}$ ).

We will show that for two indepently, and randomly chosen sparse relations with generators  $\gamma_s$ ,  $s = 1, 2$  the units  $\pm \text{UNIT}(\gamma_s, A, T)$  generate the full unit group with probability  $1/2$ .

Let  $\text{Log UNIT}(\gamma_s, A, T) = a_s R$ . Then  $\langle \pm \text{UNIT}(\gamma_s, A, T) \rangle = \mathcal{O}^*$  is equivalent to  $\gcd(c_1, c_2) = 1$ . We will first give size limits for the  $a_i$ , and then estimate the probability that the two  $a_s$  are co-prime.

**Lemma 9.** *If  $\text{Log UNIT}(\gamma_s, A, \mathbf{w}) = c_s R$ , then  $\log a_s < m \log \Delta(1 + o(1))$ .*

*Proof.* This is a consequence of (1) and  $\text{Log } \alpha_j < \Delta \log \Delta$  which holds by construction.

**Lemma 10.** *Let  $A, B, M \in \mathbb{Z}$  with  $0 < \log|A - B| < M/100$ . Consider the set  $S = \{(x, y) \in \mathbb{Z}^2 \mid A \leq x < A + M, B \leq y < B + M\}$ . If  $0 \ll M$  then there are more than  $M^2/2$  pairs  $(x, y) \in S$  with  $\gcd(x, y) = 1$ .*

*Proof.* We define the following subsets of  $S$ :

$$\begin{aligned} T &= \{(x, y) \in S \mid \gcd(x, y) \neq 1\}, \\ T_p &= \{(x, y) \in S \mid p \mid \gcd(x, y)\} \end{aligned}$$

where  $p$  denotes some prime number. We need to show that  $\text{card } T < M^2/2$ . We will show instead that

$$\sum_{p \leq M} \text{card } T_p + \text{card} \bigcup_{p > M} T_p$$

which is certainly sufficient. Note that for any two  $p, q > M$  the sets  $T_p$  and  $T_q$  are disjoint.

Let  $p \leq M$ . Then a simple counting argument shows that  $\text{card } T_p < (1 + \lfloor M/p \rfloor)^2$ . Thus

$$\begin{aligned} \sum_{p \leq M} \text{card } T_p &< \sum_{p \leq M} (1 + M/p)^2 \\ &< M(\log \log M + O(1)) + M^2 P(2), \end{aligned}$$

where  $P$  is the prime zeta function, and  $P(2) = 0.452\dots$

Let  $p > M$ . Then  $\text{card } T_p \leq 1$ . For any  $d \in \mathbb{Z}$  we define yet another set  $U_d = \{(x, y) \in S \mid x - y = d\}$ . If  $T_p \cap U_d \neq \emptyset$  then  $p \mid d$ . Thus since  $|d| < |A - B| + M$

$$\text{card}(U_d \cap \bigcup_{p > M} T_p) < \log(|A - B| + M).$$

From this we deduce

$$\begin{aligned} \text{card} \bigcup_{p > M} T_p &= \sum_{d=A-B-M}^{A-B+M} \text{card}(U_d \cap \bigcup_{p > M} T_p) \\ &< 2M(\log(|A - B| + M)) < M^2/50 + M \log M. \end{aligned}$$

Adding the two estimates we obtain the desired result for sufficiently large  $M$ .

**Corollary 3.** *Let  $(\mathbf{v}_s, \gamma_s)$  for  $s = 1, 2$  be the output of two independent calls to  $\text{RANDOMRELATION}(\mathcal{G}, (1))$  with  $\mathbf{v}_s = A\mathbf{w}_s$  for some  $\mathbf{w}_s$ . Let  $\text{Log UNIT}(\gamma_s, A, \mathbf{w}_s) = c_s R$ . Then the probability that  $\gcd(c_1, c_2) = 1$  exceeds  $(1 - o(1))1/2$ .*

*Proof.* Keep the notation from the corollary. For  $s = 1, 2$ , we fix two exponent vectors  $\mathbf{a}_s$ , and two  $c$ -smooth reduced ideals  $\mathfrak{b}_s$  in the ideal classes represented by the power products  $\mathbf{a}_s = \prod \mathfrak{p}^{a_{p,s}}$ . It suffices to show that the probability that  $\gcd(r_1, r_2) = 1$  conditional on the event that during the calls to  $\text{RANDOMRELATION}$  those exponent vectors, and ideals were chosen exceeds  $1/2$ .

The ideals  $\mathfrak{a}_s, \mathfrak{b}_s$  uniquely determine  $\mathbf{w}_s = (w_j^{(s)})$  such that  $\text{UNIT}(\gamma_s, A, \mathbf{w}_s) = \gamma_s / \beta_s$  where  $\beta_s = \prod \mathfrak{a}_j^{w_j^{(s)}}$ .

If  $(\mathfrak{c}, \gamma)$  is any of the possible values of  $\text{RANDOMRELATION}$  under the set condition then any other can be written as  $(\mathfrak{c}, \gamma')$  with  $\gamma' = \gamma \epsilon^k$  where  $\epsilon$  is the fundamental unit of  $\mathcal{O}$ , and  $k$  varies in an interval of width  $\Delta \log \Delta / R$ . Thus  $a_s = A_s + x_s$  with fixed  $A_s$ , and  $x_s < \Delta$ .

Lemma 9 implies  $A_s < m \log \Delta (1 + o(1))$ . Since  $\log m \ll \Delta$  by Corollary 2 we can apply Lemma 10. We conclude that half the pairs  $(x_1, x_2)$  yield  $\gcd(a_1, a_2) = \gcd(A_1 + x_1, A_2 + x_2) = 1$ .

Now, the lower bound for the conditional probability that a particular  $x_s$  is chosen given by Proposition 2 implies the claim.

## 7 Conclusion

In this section we give listings of RQR and RQCLR, and conclude the proof of Theorem 1.

We analyse the probability with which RQCLR produces correct output. Corollary 2 assures that steps 5 through 9 produce a matrix  $A$  whose column space equals  $L'$  with probability exceeding  $1/2$ . We obtain an approximation to  $R$  in steps 12 through 15 with the probability exceeding  $1/4$  according to Corollary 3.

Next, we assure ourselves that RQCLR never returns incorrect results.  $\tilde{h}$  computed in step 10 is always a multiple of the class number even when the previous steps yielded an  $A$  which is not a full relation matrix.

Likewise,  $\epsilon_1, \epsilon_2$  computed in step 14 are always units since they are quotients of two generators of the same ideal. So  $\tilde{R} = \gcd(\text{Log } \epsilon_1, \text{Log } \epsilon_2)$  computed approximately in step 15 is a multiple of  $R$ . Thus, step 16 assures that  $\tilde{h} = h$ , and  $\tilde{R} \approx R$ , and the precision is ensured by Maurer's algorithm.

The same argument implies that  $\tilde{R}$  obtained by RQR in each round is an approximation to a multiple of the regulator.

By Lemma 6, and Corollary 3 this multiple is the regulator itself with probability exceeding  $1/4$ . Hence, after the execution of  $O(\log(1/(1-p)))$  rounds, the minimum of all  $\tilde{R}$  computed will be an approximation to  $\mathcal{R}_\Delta$  with probability  $p$ .

Finally, we verify the time, and space complexity bound of Theorem 1. Due to Lemma 6, we need to call  $\text{RANDOMRELATION}$  in RQR  $m + 2m \log_2 \Delta (1 + o_3) =$

---

**Algorithm 1:** Probabilistic field invariant computation

---

**Input:** Discriminant  $\Delta$ 
**Output:** Class number  $h$ , elementary divisors  $s_i$  of  $Cl_\Delta$ , regulator approximation  $R'$ 


---

RQCLR( $\Delta, \epsilon$ )

1. *Validation interval* Find  $a, b$  such that  $a < hR < b < 2a$  through approximation of  $\sqrt{\Delta}L(1, \chi_\Delta)$ .
  2. *Parameters* Let  $z \leftarrow 1/\sqrt{8}$ ,  $c \leftarrow L_\Delta(\frac{1}{2}, z)$ , and  $n \leftarrow L(\frac{1}{2}, z + o_2(\Delta))$ .
  3. *Factor base* Compute and store all prime ideals in  $\mathcal{F}_c$ . Let  $m \leftarrow \text{card } \mathcal{F}$ .
  4. *Generating set* Let  $k \leftarrow 6 \log^2 \Delta$ ,  $\mathcal{G} = \mathcal{F}_k$ , and  $l \leftarrow \text{card } \mathcal{G}$ .
  5. *Full rank relation lattice* **for**  $i = 1$  **to**  $m$
  6.      $(\mathbf{v}_i, \alpha_i) \leftarrow \text{RANDOMRELATION}(\mathcal{G}, \mathfrak{p}_i^{2m\Delta})$
  7. *Full relation lattice* **for**  $j = 1$  **to**  $m$
  8.      $(\mathbf{v}_{m+j}, \alpha_{m+j}) \leftarrow \text{RANDOMRELATION}(\mathcal{F}, (1))$
  9.  $A \leftarrow (\mathbf{v}_j)_{j=1}^{n+m}$ .
  10. *Class number* Compute  $\tilde{h} \leftarrow \text{DETSS}(A)$ .
  11. *HNF* Compute with Hafner and McCurley's algorithm  $H \leftarrow \text{HNF}(A, \tilde{h})$ .
  12. *Units* Call  $\text{RANDOMRELATION}(\mathcal{G}, (1))$  twice. Let  $(\mathbf{v}_s, \gamma_s)$  be the resulting relations.
  13.      $\mathbf{w}_s \leftarrow \text{DIOPHANTINESOLVER}(A, \mathbf{v}_s)$ .
  14.     Compute  $\epsilon_s = \text{UNIT}(\gamma_s, A, \mathbf{w}_s)$ .
  15.     Compute the real GCD  $\tilde{R}$  of  $(\text{Log } \epsilon_1, \text{Log } \epsilon_2)$  using algorithm `rgcd_cfrac` in [Mau00].
  16. *Verification* **if**  $\tilde{h}\tilde{R} \notin (a, b)$  **then return** FAILURE
  17.  $h \leftarrow \tilde{h}, R' \leftarrow \tilde{R}$ .
  18. *Class group* Compute the Smith Normal Form of  $H$  which yields the elementary divisors  $d_i$  of  $Cl_\Delta$ .
  19. **return**  $(h, R', (d_i)_{i=1}^l)$ .
-

---

**Algorithm 2:** Probabilistic regulator computation
 

---

**Description:** Monte-Carlo algorithm for the computation of the regulator of a real-quadratic field

**Input:** Discriminant  $\Delta$ , error probability  $p$

**Output:** regulator approximation  $R'$  with  $|R' - \mathcal{R}_\Delta| < 1$

---

RQR( $\Delta, \epsilon$ )

1. *Parameters* Let  $z \leftarrow 1/\sqrt{8}$ ,  $c \leftarrow L_\Delta(\frac{1}{2}, z)$ , and  $n \leftarrow 2L(\frac{1}{2}, z) \log_2 \Delta(1 + o_1(\Delta))$ .
  2. *Factor base* Compute and store all prime ideals in  $\mathcal{F}_c$ . Let  $m \leftarrow \text{card } \mathcal{F}$ .
  3. *Generating set* Let  $k \leftarrow 6 \log^2 \Delta$ ,  $\mathcal{G} = \mathcal{F}_k$ , and  $l \leftarrow \text{card } \mathcal{G}$ .
  4. *Full rank relation lattice* **for**  $i = 1$  **to**  $m$
  5.      $(\mathbf{v}_i, \alpha_i) \leftarrow \text{RANDOMRELATION}(\mathcal{G}, \mathfrak{p}_i^{2m\Delta})$
  6. *Relation sequence* **for**  $j = 1$  **to**  $m$
  7.      $(\mathbf{v}_{m+j}, \alpha_{m+j}) \leftarrow \text{RANDOMRELATION}(\mathcal{F}, (1))$
  8.  $\mathcal{V} \leftarrow \{\mathbf{v}_j \mid j = 1, \dots, n+m\}$ .  
Set  $r \leftarrow 0$  and **repeat**
  9.     Set  $r \leftarrow r + 1$ . Choose randomly  $m < j_1, j_2 \leq m+n$ .
  10.    Let  $\mathbf{x}_s \leftarrow \mathbf{v}_{j_s}$  for  $s = 1, 2$
  11.    Let  $A = (\mathbf{v}_j \mid j \neq j_1, j_2)$ .
  12.     $\mathbf{w}_s \leftarrow \text{DIOPHANTINE SOLVER}(A, \mathbf{v}_s)$  for  $s = 1, 2$
  13.    Compute  $\epsilon_s = \text{UNIT}(\gamma_s, A, \mathbf{w}_s)$  for  $s = 1, 2$ .
  14.    Compute the real GCD  $\tilde{R}$  of  $(\text{Log } \epsilon_1, \text{Log } \epsilon_2)$  using algorithm `rgcd_cfrac` in [Mau00].
  15.     $R' \leftarrow \min(R', \tilde{R})$ .
  16. **until**  $(3/4)^{r-1} < p$
  17. **return**  $R'$ .
-

$L(\frac{1}{2}, z + o(1))$  times. Each call takes estimated time bounded by  $L(\frac{1}{2}, 1/(4z) + o(1))$ . In RQCLR we need  $L(\frac{1}{2}, z + o_4)$  relations, but this time each call to RANDOMRELATION costs time  $L(\frac{1}{2}, z + 1/(4z))$  due to the longer time needed to compute the random power product. Note that the estimated time needed for the factorisations in RANDOMRELATION can be subsumed into the  $o(1)$  term, cf. [LP92].

The solution of the two Diophantine systems to obtain the redundant relations takes time  $L(\frac{1}{2}, 3z + o(1))$ . The remaining steps needed to arrive at the regulator multiple take only time  $L(\frac{1}{2}, 2z + o(1))$  due to Lemma 9, and Theorem 12.1.5 of [Mau00].

The optimum run time of both algorithms will be achieved with  $z = 1/\sqrt{8}$  which yields the run time bounds of Theorem 1, and concludes the proof of the theorem.  $\square$

As was already shown in [HM89], the determinant of  $L'$ , and its primary invariants can be computed in time  $O((n + m)^4)$ . Indeed, the exponent can be lowered to 3 by combining Vollmer's algorithm for the computation of the determinant of the column space of a matrix with small essential part, see [Vol00], with Hafner and McCurley's HNF algorithm proposed in [HM91].

Since in RQCLR the time needed for the linear algebra part of the algorithm is majorised by the time needed for the relation generation, it remains to be shown that sparse relations are sufficient for the generation of the full integral relation lattice  $L'$ . This would allow to employ the fast HNF algorithm to full advantage, and is a task for future research.

## References

- [Abe94] Christine Abel. *Ein Algorithmus zur Berechnung der Klassenzahl und des Regulators reellquadratischer Ordnungen*. PhD thesis, Universität des Saarlandes, Saarbrücken, Germany, 1994. German.
- [Bac90] Eric Bach. Explicit bounds for primality testing and related problems. *Math. Comp.*, 55(191):355–380, 1990.
- [BTW95] Johannes Buchmann, Christoph Thiel, and Hugh C. Williams. Short representation of quadratic integers. In Wieb Bosma and Alf J. van der Poorten, editors, *Computational Algebra and Number Theory, Sydney 1992*, volume 325 of *Mathematics and its Applications*, pages 159–185. Kluwer Academic Publishers, 1995.
- [Buc90] Johannes Buchmann. A subexponential algorithm for the determination of class groups and regulators of algebraic number fields. In Catherine Goldstein, editor, *Séminaire de Théorie des Nombres, Paris 1988–1989*, volume 91 of *Progress in Mathematics*, pages 27–41. Birkhäuser, 1990.
- [Buc89] J. Buchmann. A subexponential algorithm for the determination of class groups and regulators of algebraic number fields. In *Séminaire de Théorie des Nombres*, pages 27–41, Paris, 1988–89.
- [Coh93] H. Cohen. *A course in computational algebraic number theory*. Springer, Heidelberg, 1993.
- [HM89] James L. Hafner and Kevin S. McCurley. A rigorous subexponential algorithm for computation of class groups. *J. Am. Math. Soc.*, 2(4):837–850, 1989.

- [HM91] J.L. Hafner and K.S. McCurley. Asymptotically fast triangularization of matrices over rings. *SIAM J. Comput.*, 20:1068–1083, 1991.
- [Jac99] Michael J. Jacobson, Jr. Applying sieving to the computation of quadratic class groups. *Mathematics of Computation*, 68(226):859–867, 1999.
- [Len82] Hendrik W. Lenstra, Jr. On the calculation of regulators and class numbers of quadratic fields. In J. V. Armitage, editor, *Journées Arithmétiques, Exeter 1980*, volume 56 of *London Mathematical Society Lecture Notes Series*, pages 123–150. Cambridge University Press, 1982.
- [LP92] H.W. Lenstra Jr. and C. Pomerance. A rigorous time bound for factoring integers. *J. Amer. Math. Soc.*, 5:483–516, 1992.
- [Mau00] Markus Maurer. *Regulator approximation and fundamental unit computation for real quadratic orders*. PhD thesis, Technische Universität Darmstadt, Fachbereich Informatik, Darmstadt, Germany, 2000.
- [MS99] Th. Mulders and A. Storjohann. Diophantine linear system solving. In *Proc. Int'l. Symp. on Symbolic and Algebraic Computation: ISSAC '99*, 1999. to appear.
- [Sey87] Martin Seysen. A probabilistic factorization algorithm with quadratic forms of negative discriminant. *Mathematics of Computation*, 48:757–780, 1987.
- [Vol00] Ulrich Vollmer. Asymptotically fast discrete logarithms in quadratic number fields. In Wieb Bosma, editor, *Algorithmic Number Theory Symposium IV*, volume 1838 of *Lecture Notes in Computer Science*, pages 581–594. Springer-Verlag, 2000.