



Centrum voor Wiskunde en Informatica

**REPORT**RAPPORT

MAS

Modelling, Analysis and Simulation



*Modelling, Analysis and Simulation*

New computations concerning the Cohen-Lenstra  
heuristics

H.C. Williams, H.J.J. te Riele

**REPORT MAS-R0215 JUNE 30, 2002**

CWI is the National Research Institute for Mathematics and Computer Science. It is sponsored by the Netherlands Organization for Scientific Research (NWO).

CWI is a founding member of ERCIM, the European Research Consortium for Informatics and Mathematics.

CWI's research has a theme-oriented structure and is grouped into four clusters. Listed below are the names of the clusters and in parentheses their acronyms.

Probability, Networks and Algorithms (PNA)

Software Engineering (SEN)

**Modelling, Analysis and Simulation (MAS)**

Information Systems (INS)

Copyright © 2001, Stichting Centrum voor Wiskunde en Informatica

P.O. Box 94079, 1090 GB Amsterdam (NL)

Kruislaan 413, 1098 SJ Amsterdam (NL)

Telephone +31 20 592 9333

Telefax +31 20 592 4199

ISSN 1386-3703

# New Computations Concerning the Cohen-Lenstra Heuristics

Hugh Williams

*Department of Mathematics and Statistics, University of Calgary, Calgary, Alberta, Canada T2N 1N4*  
*williams@math.ucalgary.ca*

Herman te Riele

*CWI, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands*  
*herman@cwi.nl*

## ABSTRACT

Let  $h(p)$  denote the class number of the real quadratic field formed by adjoining  $\sqrt{p}$ , where  $p$  is a prime, to the rationals. The Cohen-Lenstra heuristics suggest that the probability that  $h(p) = k$  (a given odd positive integer) is given by  $Cw(k)/k$ , where  $C$  is an explicit constant and  $w(k)$  is an explicit arithmetic function. For example, we expect that about 75.45% of the values of  $h(p)$  are 1, 12.57% are 3, and 3.77% are 5. Furthermore, a conjecture of Hooley states that

$$H(x) := \sum_{p \leq x} h(p) \sim x/8 \text{ as } x \rightarrow \infty,$$

where the sum is taken over all primes congruent to 1 modulo 4. In this paper, we develop some fast techniques for evaluating  $h(p)$  where  $p$  is not very large and provide some computational results in support of the Cohen-Lenstra heuristics. We do this by computing  $h(p)$  for all  $p \equiv 1 \pmod{4}$  and  $p < 2 \times 10^{11}$ . We also tabulate  $H(x)$  up to  $2 \times 10^{11}$ .

*2000 Mathematics Subject Classification:* Primary 11R29. Secondary 11Y40.

*1998 ACM Computing Classification System:* F.2.1.

*Keywords and Phrases:* Distribution of class numbers, Cohen-Lenstra heuristics, Hooley's conjecture.

*Note:* The research of Williams was partially supported by NSERC of Canada Grant #A7649. The research of Te Riele was carried out under project MAS2.2 "Computational number theory and data security".

## 1. INTRODUCTION

Let  $D$  denote a square-free positive integer and let  $\mathcal{K} = \mathbb{Q}(\sqrt{D})$  be the quadratic field formed by adjoining  $\sqrt{D}$  to the rationals  $\mathbb{Q}$ . Set

$$r = \begin{cases} 2 & \text{when } D \equiv 1 \pmod{4}, \\ 1 & \text{otherwise.} \end{cases}$$

If  $\omega = (r - 1 + \sqrt{D})/r$ , then  $\mathcal{O} = \mathbb{Z} + \omega\mathbb{Z}$  is the maximal order (the ring of algebraic integers) of  $\mathcal{K}$ . Let  $\epsilon (> 1)$  be the fundamental unit of  $\mathcal{K}$ ,  $R = \log \epsilon$  be the regulator of  $\mathcal{K}$  and  $h = h(D)$  be the class number of  $\mathcal{K}$ .

In [2, 3], Cohen and Lenstra developed some heuristics to explain the distribution of the odd part of the class groups of quadratic fields. In particular, they gave reasons to expect that the probability that  $h^*(D)$  (the odd part of  $h(D)$ ) is equal to a given positive odd integer  $k$  is given by

$$\text{Prob}(h^*(D) = k) = Cw(k)/k, \tag{1.1}$$

where  $C = .754458173\dots$  and

$$w(k)^{-1} = \prod_{p^\alpha \parallel k} p^\alpha (1 - p^{-1}) (1 - p^{-2}) \dots (1 - p^{-\alpha}).$$

If  $D$  is a prime, then  $h^*(D) = h(D)$ . Also, it is not unreasonable to expect that quadratic fields with prime values of  $D$  behave like any others with respect to the odd part of the class group; thus, we would expect that

$$\text{Prob}(h(p) = 1) = C,$$

when  $p$  is a prime. This suggests that for at least  $3/4$  of all primes we have  $h(p) = 1$ ; it must, however, be stressed here that it is not even known that there exists an infinitude of values of  $D$  for which  $h(D) = 1$ . Nevertheless, computations performed by Stephens and Williams [11], Jacobson, Lukes, and Williams [6], and Jacobson [5] provide much numerical evidence in support of the Cohen-Lenstra heuristics.

We also mention that with some additional assumptions Cohen was able to show (assuming the Cohen-Lenstra heuristics) that

$$H(x) := \sum_{\substack{p \leq x \\ p \equiv 1 \pmod{4}}} h(p) \sim x/8,$$

a result conjectured by Hooley [4]. This conjecture and (1.1) were tested for all primes  $p \equiv 1 \pmod{4}$  up to  $10^9$  in [6]. It was found that  $H(x)/x$  seemed to be increasing at such a slow rate that it is hard to predict whether it would reach  $1/8$ , but that for small values of  $k$ , (1.1) gives a quite accurate prediction of what actually happens for  $p < 10^9$ .

In Van der Poorten, Te Riele, and Williams [9], some very fast methods were developed for computing in real quadratic fields when  $D$  is not very large. These were used to verify the Ankeny-Artin-Chowla conjecture for all primes  $p (\equiv 1 \pmod{4})$  such that  $p < 10^{11}$ . In this paper we will show how these ideas can be extended to the problem of testing the Cohen-Lenstra heuristics for the same (and also larger) values of  $p$  and for testing Hooley's conjecture for these  $p$ . As there are 4 003 548 492 primes congruent to 1 modulo 4 up to  $2 \times 10^{11}$ , it was necessary to develop very fast methods to compute  $h(p)$  for  $p$  in this range.

We make use of the analytic class number formula

$$2h(p)R = \sqrt{p} L(1, \chi_p), \tag{1.2}$$

where  $L(1, \chi_p)$  is the Dirichlet  $L$ -function of the character  $\chi_p$  evaluated at  $s = 1$ . We will let  $R_2 = \log_2 \epsilon = (\log_2 e)R$ . We will also assume the truth of the extended Riemann hypothesis (ERH) for  $L(s, \chi_p)$ . Broadly speaking our algorithm to compute  $h(p)$  consists of two main components:

1. Computation of  $R_2$

- (a) Find an integral multiple  $M$  of  $R_2$ . This step is fully described in [9].
- (b) Compute  $R_2$  from  $M$  or prove that  $R_2 > M/P$ , where  $P$  is some small prime (e.g. 11 or 13).
- (c) Given that  $R_2 > M/P$ , find  $R_2$ .

2. Find  $h = h(p)$

- (a) We use the approximation  $S(T, p)$  (for suitable  $T$ ) of  $\log L(1, \chi_p)$ , computed in Step 1a. This satisfies, on the assumption of the ERH:

$$|\log L(1, \chi_p) - S(T, p)| < A(T, p),$$

where  $A(T, p)$  is an error bound discussed in Section 3.

Let  $\text{Ne}(x)$  denote the nearest odd integer to  $x$ , and put

$$\begin{aligned}\tilde{h} &:= \text{Ne} \left( \frac{\sqrt{p} \exp(S(T, p))}{R_2 \log 4} \right) \in \mathbb{N}, \\ \delta &:= \frac{\sqrt{p} \exp(S(T, p))}{R_2 \log 4} - \tilde{h} \quad (|\delta| < 1).\end{aligned}$$

(b) Try to compute  $h$  from  $\tilde{h}$ .

Put  $h_1 = 1$ .

Suppose  $h_1 \geq |g - \delta|/2$ , where  $g \in \mathbb{Z}$ .

If  $\tilde{h} + g = h_1$  and

$$\exp(A(T, p)) < 3h_1/(\tilde{h} + \delta), \quad (1.3)$$

then  $h = \tilde{h} + g$ .

If  $\tilde{h} + g \geq 3h_1$  and

$$\exp(A(T, p)) < \min \left\{ \frac{\tilde{h} + g + 2h_1}{\tilde{h} + \delta}, \frac{\tilde{h} + \delta}{\tilde{h} + g - 2h_1} \right\}, \quad (1.4)$$

then  $h = \tilde{h} + g$ .

If this procedure does not find  $h$ , then find some  $h_1 > 1$  such that  $h_1 | \tilde{h} + g$ ,  $h_1 | h$ ,  $h_1 > |g - \delta|/2$  and try again.

(c) If  $h$  can not be found in Step 2b, treat it as a separate case, to be dealt with later.

Evidently, this method is a variant of Lenstra's [8] algorithm for evaluating  $R$  and  $h(p)$ . This is of computational complexity  $\mathcal{O}(p^{1/5+\epsilon})$  under the ERH. What we need to do here is make the process execute as rapidly as possible for values of  $p$  that are relatively small, in our case  $p < 2 \times 10^{11}$ .

## 2. DETERMINATION OF THE REGULATOR $R_2$ FROM AN INTEGRAL MULTIPLE $M$ OF $R_2$

For the sake of brevity we will make use of the same notation as that used in [9] as well as several results used there. If  $\mathfrak{b}$  is any reduced principal integral ideal of  $\mathcal{O}$ , we let

$$\mathfrak{b}_1 (= \mathfrak{b}), \mathfrak{b}_2, \mathfrak{b}_3, \dots, \mathfrak{b}_m, \dots \quad (2.1)$$

be the sequence of reduced principal ideals produced by applying the continued fraction algorithm to  $\mathfrak{b}$  (see [9]). We let  $\Psi_1 = 1$  and

$$\Psi_j = \prod_{i=1}^{j-1} \psi_i$$

have the same meaning as that assumed in [9] and we have  $\mathfrak{b}_j = (\Psi_j)\mathfrak{b}_1$ . We define  $\zeta_j = \zeta(\mathfrak{b}_j)$ ,  $\rho_j = \rho(\mathfrak{b}_j)$  by

$$2^{\zeta_j-1} < \Psi_j < 2^{\zeta_j}, \rho_j = 2^{\zeta_j}/\Psi_j.$$

**Lemma 2.1**  $1 < \rho_j < 2$ .

**Proof.** Follows easily from the definition of  $\rho_j$ . □

**Lemma 2.2** If  $\mathfrak{b}_1 = (1)$  and  $m$  is the least positive integer ( $> 1$ ) such that  $\mathfrak{b}_m = (1)$ , then

$$R_2 = \zeta_m - \log_2 \rho_m.$$

**Proof.** Follows from the fact that  $\Psi_m = \epsilon$  and the definition of  $\zeta_j$  and  $\rho_j$ .  $\square$

If  $x \geq 0$  is a real number, we define  $\mathfrak{b}(x)$  to be that ideal in the sequence (2.1) such that  $\Psi_j \leq 2^x$  and  $\Psi_{j+1} > 2^x$ . We also define  $\rho(x) = 2^x/\Psi_j$ .

Let  $B = \lceil \log_2(2\sqrt{D}/r) \rceil$  and recall from [9] that  $\log_2(L(\mathfrak{b}_i)\psi_i) < B$  and  $\log_2(L(\mathfrak{b}(x))\rho(x)) < B$ . Here  $L(\mathfrak{b})$  denotes the least positive rational integer in the ideal  $\mathfrak{b}$ . If  $\mathfrak{b}$  is a reduced ideal, then  $L(\mathfrak{b}) = N(\mathfrak{b})$ , where  $N(\mathfrak{b})$  is the norm of  $\mathfrak{b}$ . Let  $t$  be any positive real such that  $t \geq 2B + 1$  and let  $\mathcal{L}$  be the list of ideals

$$\{\mathfrak{b}_1, \mathfrak{b}_2, \dots, \mathfrak{b}_{m-1}\}, \quad (2.2)$$

where  $m$  is the least positive integer such that  $\zeta_m > t + B + 1$ . Assume that  $\mathfrak{b}_1$  is the only ideal  $\mathfrak{b}$  in  $\mathcal{L}$  such that  $\mathfrak{b} = (1)$ . Under these circumstances we have the following Lemma and Theorems.

**Lemma 2.3**  $\epsilon > 2^t$ .

**Proof.** We know that  $\epsilon = \Psi_r$  and  $r \geq m$ , so that  $\epsilon = \Psi_r \geq \Psi_m > 2^{\zeta_m - 1} > 2^{t+B}$ .  $\square$

**Theorem 2.4** *There must exist some  $i \geq 1$  such that either  $\mathfrak{b}(2it) \in \mathcal{L}$  or  $\bar{\mathfrak{b}}(2it) \in \mathcal{L}$ .*

**Proof.** Since  $\epsilon > 2^t$ , there is a unique  $n \in \mathbb{Z}$  such that  $n \geq 2$  and

$$2^{(n-1)t} < \epsilon < 2^{nt}.$$

If  $2|n$ , put  $i = n/2$  and  $\mathfrak{b}_k = \mathfrak{b}(nt) = \mathfrak{b}(2it)$ . Here, we may assume that  $\mathfrak{b}_k = (\Psi_k)$  where  $\Psi_k \leq 2^{nt}$ ,  $\Psi_{k+1} > 2^{nt}$ . It follows that since  $\epsilon = \Psi_r$  and  $\Psi_r < 2^{nt}$ , we must have  $r \leq k$ ; hence,  $\epsilon \leq \Psi_r$ . If we consider  $\theta = \Psi_k \epsilon^{-1}$ , we have  $\theta \geq 1$  and

$$\theta < 2^{-(n-1)t} 2^{nt} = 2^t.$$

Since  $\mathfrak{b}_k$  is a reduced ideal, so is  $(\theta) (= \mathfrak{b}_k)$ . Hence,  $(\theta) = \mathfrak{b}_j$  and  $\mathfrak{b}_j = (\Psi_j)$ , where

$$1 \leq \Psi_j < 2^t \Rightarrow 2^{\zeta_j - 1} < 2^t \Rightarrow \zeta_j < t + 1 \Rightarrow \mathfrak{b}_j \in \mathcal{L}.$$

If  $2 \nmid n$ , put  $i = (n-1)/2$  and  $\mathfrak{b}_k = \mathfrak{b}(2it)$ . Now consider  $\theta = \epsilon |\bar{\Psi}_k|$ . We know that  $\Psi_k |\bar{\Psi}_k| = L(\mathfrak{b}(2it)) := L_k \in \mathbb{Z}^+$ . Hence,

$$\theta = \epsilon L_k / \Psi_k > \frac{2^{(n-1)t} L_k}{2^{2it}} = L_k > 1.$$

Also,

$$\theta < \frac{2^{nt} L_k}{2^{2it}} \psi_k < 2^t (2\sqrt{D}/r). \quad (L_k \psi_k < 2\sqrt{D}/r)$$

Now  $(\theta) = \bar{\mathfrak{b}}_k$  is reduced; thus,  $(\theta) = \mathfrak{b}_j = (\Psi_j)$  and  $1 < \Psi_j < 2^t (2\sqrt{D}/r)$ . Also,

$$2^{\zeta_j - 1} < \Psi_j \Rightarrow \zeta_j - 1 < t + B \Rightarrow \zeta_j < t + B + 1 \Rightarrow \mathfrak{b}_j \in \mathcal{L}.$$

$\square$

**Theorem 2.5** *Let  $i$  be the least integer ( $\geq 1$ ) such that either  $\mathfrak{b}(2it) \in \mathcal{L}$  or  $\bar{\mathfrak{b}}(2it) \in \mathcal{L}$ .*

*If  $\mathfrak{b}(2it) \in \mathcal{L}$  and  $\mathfrak{b}(2it) = \mathfrak{b}_j$ , then*

$$R_2 = 2it - \zeta_j - \log_2(\rho(2it)/\rho_j).$$

*If  $\bar{\mathfrak{b}}(2it) \in \mathcal{L}$  and  $\bar{\mathfrak{b}}(2it) = \mathfrak{b}_j$ , then*

$$R_2 = 2it + \zeta_j - \log_2(\rho(2it)\rho_j L(\mathfrak{b}(2it))).$$

**Proof.** As before, define  $n$  ( $\geq 2$ ) by

$$2^{(n-1)t} < \epsilon < 2^{nt}.$$

We use the same notation as in the previous theorem. Put

$$i' = \begin{cases} n/2 & \text{if } 2|n \\ (n-1)/2 & \text{if } 2 \nmid n. \end{cases}$$

We know that either  $\mathfrak{b}(2i't)$  or  $\overline{\mathfrak{b}}(2i't) \in \mathcal{L}$  by the previous theorem. Hence  $i \leq i'$ . If  $i = i'$ , then  $\epsilon = \Psi_k/\Psi_j$  when  $\mathfrak{b}(2i't) = \mathfrak{b}_j \in \mathcal{L}$ , or  $\epsilon = \Psi_k\Psi_j/L_k$  when  $\overline{\mathfrak{b}}(2i't) = \mathfrak{b}_j \in \mathcal{L}$ . Thus, we may assume that  $i \leq i' - 1 \Rightarrow n - 1 \geq 2i$ .

If  $\mathfrak{b}(2it) \in \mathcal{L}$ , then  $\eta = \Psi_k/\Psi_j \leq 2^{2it}$  is a unit and

$$\eta = \frac{\Psi_k}{\Psi_j} > \frac{2^{2it}}{\psi_k\Psi_j} > 2^{2it-B-\zeta_j} \geq 2^{2t-B-\zeta_j} > 2^{2t-B-(t+B+1)} = 2^{t-2B-1} \geq 1.$$

Thus,  $\eta = \epsilon^l$  ( $l \geq 1$ ). If  $l = 1$ , we are done. If  $l > 1$ , then  $\eta \geq \epsilon^2$  and

$$2^{2it} \geq 2^{2(n-1)t} \geq 2^{4it}, \text{ a contradiction.}$$

If  $\overline{\mathfrak{b}}(2it) \in \mathcal{L}$ , then  $\eta = \Psi_k\Psi_j/L_k$  is a unit and

$$\eta = \Psi_k\Psi_j/L_k > \Psi_j 2^{2it}/L_k\psi_k > 2^{2it-B} > 1.$$

Again, we have  $\eta = \epsilon^l$  ( $l \geq 1$ ). If  $l > 1$ , then  $\eta \geq \epsilon^2$  and

$$\Psi_j 2^{2it} \geq \eta \geq 2^{2(n-1)t} \geq 2^{4it}.$$

Since  $\Psi_j < 2^{\zeta_j} < 2^{t+B+1}$ , we get  $t \leq B+1$ , a contradiction. Thus, in the first case we get

$$\epsilon = \Psi_k/\Psi_j = 2^{2it} \frac{1}{\rho(2it)} \Big/ 2^{\zeta_j} \frac{1}{\rho_j} \Rightarrow R_2 = 2it - \zeta_j - \log_2(\rho(2it)/\rho_j).$$

In the second case we get

$$\epsilon = \Psi_k\Psi_j/L_k \Rightarrow R_2 = 2it + \zeta_j - \log_2(\rho(2it)\rho_j L(\mathfrak{b}(2it))).$$

□

The following corollary to Theorem 2.5 will be useful in a subsequent section.

**Corollary 2.6** *If  $n, i, i'$  are defined as in the Theorem, then  $i = i'$  when  $2 \nmid n$ , and  $i = i'$  or  $i' - 1$  if  $2|n$ .*

**Proof.**

**Case 1.**  $2|n$ .

In this case we have  $n = 2i'$  and  $\epsilon \geq 2^{(2i'-1)t}$ . Now if  $\epsilon = \Psi_k/\Psi_j$ , we get

$$\epsilon \leq \Psi_k \leq 2^{2it}.$$

It follows that  $2it \geq (2i' - 1)t$ ; hence,

$$2i \geq 2i' - 1 \Rightarrow i \geq i' \Rightarrow i = i'$$

(recall that  $i \leq i'$ ). If  $\epsilon = \Psi_k \Psi_j / L_k$ , then

$$\epsilon < 2^{2it+t+B+1}.$$

We get

$$(2i' - 1)t < (2i + 1)t + B + 1$$

and

$$2i' - 1 < 2i + 1 + \frac{B+1}{t} < 2i + 2.$$

Thus,

$$2i' - 1 \leq 2i + 1$$

and

$$i' \leq i + 1 \Rightarrow i = i' \text{ or } i' - 1.$$

**Case 2.**  $2 \nmid n$ .

In this case we have  $n = 2i' + 1$  and  $\epsilon > 2^{2i't}$ . If  $\epsilon = \Psi_k \Psi_j / L_k$ , then

$$2it + t + B + 1 > 2i't$$

and

$$2i + 2 > 2i' \Rightarrow i + 1 > i' \Rightarrow i \geq i' \Rightarrow i = i'.$$

If  $\epsilon = \bar{\Psi}_k / \bar{\Psi}_j$ , then

$$2^{2it} > \epsilon \geq 2^{2i't} \Rightarrow i \geq i' \Rightarrow i = i'.$$

□

We can now make use of the following algorithms to find  $R_2$ , given an integral multiple  $M$  of  $R_2$ .

- 1) Select a prime  $P$  such that  $P^2 B < M$ . In our computations, we used  $P = 11$ .
- 2) Put  $K = M/P$ ,  $t = fc$  (see [9, p. 1325]).

**Algorithm 2.7** (compute  $R_2$  or prove that  $R_2 > K$ )

- 1) Compute the list  $\mathcal{L}$  (2.2). If  $\mathfrak{b}_j = (1)$  for  $\mathfrak{b}_j \in \mathcal{L}$ , compute  $R_2 = \zeta_j - \log_2 \rho_j$  and terminate.
- 2) For  $i = 1, 2, \dots, \lceil (K + 2B + 1)/2t \rceil$ , compute  $\mathfrak{b}(2it)$ .
  - If  $\mathfrak{b}(2it) = \mathfrak{b}_j \in \mathcal{L}$ , then  $R_2 = 2it - \zeta_j - \log_2(\rho(2it)/\rho_j)$  and terminate.
  - If  $\bar{\mathfrak{b}}(2it) = \mathfrak{b}_j \in \mathcal{L}$ , then  $R_2 = 2it + \zeta_j - \log_2(\rho(2it)\rho_j L(\mathfrak{b}(2it)))$  and terminate.

End for

$R_2 > K$ .

**Proof** (of correctness). Clearly, when  $R_2$  is computed, it is correct by Lemma 2.2 and Theorem 2.5. Suppose  $R_2$  is not computed by the algorithm; we know that for some  $i$  we must have either

$$R_2 = 2it - \zeta_j - \log_2(\rho(2it)/\rho_j)$$

or

$$R_2 = 2it + \zeta_j - \log_2(\rho(2it)\rho_j L(\mathfrak{b}(2it)))$$

and  $i > \lceil (K + 2B + 1)/2t \rceil$ . In the first case we have

$$R_2 > \left( \frac{K + 2B + 1}{2t} \right) 2t + 2t - (t + B + 1) - B > K.$$

In the second

$$R_2 > 2t \left( \frac{K + 2B + 1}{2t} \right) - B + 1 > K.$$



□

We let  $\{p_1(=3), p_2, p_3, \dots, p_j\}$  be the ordered set of all primes  $< P$ . Then  $p_{j+1} = P$ . We can now use the following algorithm to compute  $R_2$  when Algorithm 2.7 fails to do so.

**Algorithm 2.8** (given that  $R_2 > K$ , find  $R_2$ )

1)  $\mathfrak{b}_1 = (1), i \leftarrow 1, M' \leftarrow M$ .

2) while  $i \leq j$

    compute  $\mathfrak{b}(M'/p_i)$

    if  $\mathfrak{b}(M'/p_i) = \mathfrak{b}_1$

$M' \leftarrow M'/p_i$

    else

$i \leftarrow i + 1$

    end if

end while

$R_2 = M'$ .

**Proof** (of correctness). We first note that if  $M'$  is an integral multiple of  $R_2$ , say  $M' = sR_2$ , and  $\mathfrak{b}(M'/p) = \mathfrak{b}_1$  for some prime  $p < P$ , then  $\mathfrak{b}_k = (\Psi_k) = \mathfrak{b}(M'/p)$ , where  $\Psi_k = \epsilon^t$  ( $t \geq 0$ ),  $\epsilon^t \geq 2^{M'/p}$  and  $\epsilon^t < (2\sqrt{D}/r)2^{M'/p}$ . It follows that  $\epsilon^{pt} \geq 2^{M'} = \epsilon^s$  and  $pt \geq s$ . Furthermore, since  $\epsilon^{pt} < (2\sqrt{D}/r)^p \epsilon^s$ , we get  $(2\sqrt{D}/r)^p > \epsilon^{pt-s}$ . If  $pt - s \geq 1$ , then  $pB > R_2$  and  $PB > K = M/P$ , a contradiction. Thus we must have  $tp = s$ , which means that  $M'/p$  is an integral multiple of  $R_2$ .

We also note that at the end of the algorithm we have  $M' = sR_2, s \in \mathbb{Z}$  and  $p_i \nmid s$  for all  $i \leq j$ . Then,  $M' = R_2$  or  $s \geq p_{j+1} = P$ . Now

$$M \geq M' = sR_2 \Rightarrow R_2 \leq M/P = K,$$

a contradiction. Thus,  $R_2 = M'$ . □

### 3. A MODIFICATION OF BACH'S RESULT

In [1] Bach provided (under the ERH) explicit constants  $A, B$  such that if

$$A'(T, p) = (A \log p + B)/(\sqrt{T} \log T), \quad (3.1)$$

then

$$\left| \log L(1, \chi_p) - \sum_{i=0}^{T-1} a_i \log B(T+i) \right| < A'(T, p),$$

where  $a_i = (x+i) \log(x+i)/S(x)$ ,  $S(x) = \sum_{i=0}^{x-1} (x+i) \log(x+i)$  and  $B(x) = \prod_{q < x} (1 - \chi_p(q)/q)^{-1}$ . This allows us to get an estimate for  $L(1, \chi_p)$  which is very useful for determining  $h(p)$  once  $R_2$  has been computed. Since most of the values of  $h(p)$  tend to be small, we found it useful to try to improve Bach's results. Our improvement is only a very slight one, but it proved to be very effective for determining  $h(p)$  for many values of  $p$ . As the technique of deriving this improvement is analogous to the treatment given by Jacobson and Williams [7] for estimating  $L(2, \chi)$ , we will only sketch it here.

As in [1], we put

$$\overline{B}(x, \chi) = \prod_{q \geq x} \frac{q}{q - \chi(q)}, \quad B(x, \chi) = \prod_{q < x} \frac{q}{q - \chi(q)},$$

where the products are taken over prime values of  $q$  and  $\chi$  is a nonprincipal character modulo  $m$ . Since

$$\log L(1, \chi) = \sum_{i=0}^{x-1} a_i \log L(1, \chi) = \sum_{i=0}^{x-1} a_i B(x+i, \chi) + \sum_{i=0}^{x-1} a_i \overline{B}(x+i, \chi),$$

we need to bound the value of

$$E(x, \chi) = \sum_{i=0}^{x-1} a_i \log \bar{B}(x+i, \chi).$$

As in [1], we get

$$\begin{aligned} |E(x, \chi)| &\leq \left| \sum_{i=0}^{x-1} a_i \frac{\Psi(x+i-1, \chi)}{(x+i) \log(x+i)} \right| + \left| \sum_{i=0}^{x-1} a_i \frac{\Psi^1(x+i, \chi)(\log(x+i)+1)}{(x+i)^2 (\log(x+i))^2} \right| \\ &+ \left| \sum_{i=0}^{x-1} a_i \int_x^\infty \frac{\Psi^1(t, \chi)}{t^3} \left( \frac{2}{\log t} + \frac{3}{(\log t)^2} + \frac{2}{(\log t)^3} \right) dt \right| \\ &+ \left| \sum_{i=0}^{x-1} a_i T(x+i, \chi) \right|. \end{aligned}$$

The method of Lemma 5.1 of [1] can be used to prove that

$$|T(x, \chi)| \leq 2C \left( \frac{2}{x^{1/2} \log x} + \frac{3/2}{\log 2} x^{-2/3} \right),$$

where  $C = 1.25506$ . Hence,

$$\begin{aligned} \left| \sum_{i=0}^{x-1} a_i T(x+i, \chi) \right| &\leq 4C \sum_{i=0}^{x-1} \frac{a_i}{(x+i)^{1/2} \log(x+i)} + \frac{3C}{\log 2} x^{-2/3} \\ &\leq \frac{4C}{S(x)} \sum_{i=0}^{x-1} (x+i)^{1/2} + \frac{3C}{\log 2} x^{-2/3}. \end{aligned}$$

As noted in [7],

$$\sum_{i=0}^{x-1} (x+i)^{1/2} < \lambda x^{3/2},$$

where  $\lambda = 2(2^{3/2} - 1)/3 \approx 1.2189514$ ; hence,

$$\left| \sum_{i=0}^{x-1} a_i T(x+i, \chi) \right| \leq \frac{4C}{S(x)} \lambda x^{3/2} + \frac{3C}{\log 2} x^{-2/3}.$$

Also,

$$\begin{aligned} S(x) > U(x) &:= \int_0^{x-1} (t+x) \log(t+x) dt \\ &= \frac{1}{2} \left[ (2x-1)^2 \left( \log(2x-1) - \frac{1}{2} \right) - x^2 \left( \log x - \frac{1}{2} \right) \right]. \end{aligned}$$

Since, under the ERH, we have

$$\Psi^1(x, \chi) \leq c(m)x^{3/2} + h(x),$$

where

$$c(m) = \frac{2}{3} \left( \log m + \frac{5}{3} \right)$$

and

$$h(x) = x \log x + 2(c(m) + 1)x + 3c(m) + 1,$$

we can use the reasoning of [1] to find that:

$$\begin{aligned} \left| \sum_{i=0}^{x-1} a_i \frac{\Psi(x+i-1, \chi)}{(x+i) \log(x+i)} \right| &< \frac{(1+2^{3/2})c(m)x^{3/2}}{U(x)} + \frac{h(x)+h(2x)}{x^2 \log x}, \\ \left| \sum_{i=0}^{x-1} a_i \frac{\Psi^1(x+i, \chi)(\log(x+i)+1)}{(x+i)^2(\log(x+i))^2} \right| &\leq \frac{c(m)\lambda x^{3/2}}{U(x)} + \frac{c(m)\lambda}{x^{1/2}(\log x)^2} + \frac{h(x)(1+\log x)}{x^2(\log x)^2}, \\ \left| \sum_{i=0}^{x-1} a_i \int_x^\infty \frac{\Psi^1(t, \chi)}{t^3} \left( \frac{2}{\log t} + \frac{3}{(\log t)^2} + \frac{2}{(\log t)^3} \right) dt \right| \\ &< \frac{2c(m)\lambda x^{3/2}}{U(x)} \left( 2 + \frac{3}{\log x} + \frac{2}{(\log x)^2} \right) + \int_x^\infty \frac{h(t)}{t^3} \left( \frac{2}{\log t} + \frac{3}{(\log t)^2} + \frac{2}{(\log t)^3} \right) dt. \end{aligned}$$

We can next deduce (again using the reasoning in [1]) that

$$\begin{aligned} &\frac{h(x)+h(2x)}{x^2 \log x} + \frac{h(x)(1+\log x)}{x^2(\log x)^2} + \int_x^\infty \frac{h(t)}{t^3} \left( \frac{2}{\log t} + \frac{3}{(\log t)^2} + \frac{2}{(\log t)^3} \right) dt \\ &\leq c(m) \left[ \frac{12}{x \log x} + \frac{8}{x(\log x)^2} + \frac{4}{x(\log x)^3} + \frac{12}{x^2 \log x} + \frac{15}{2x^2(\log x)^2} + \frac{3}{x^2(\log x)^3} \right] \\ &+ \frac{6}{x} + \frac{10+2 \log 2}{x \log x} + \frac{6}{x(\log x)^2} + \frac{2}{x(\log x)^3} + \frac{4}{x^2 \log x} + \frac{5}{2x^2(\log x)^2} + \frac{1}{x^2(\log x)^3}. \end{aligned}$$

On combining our previous results, we see that

$$\left| \log L(1, \chi) - \sum_{i=0}^{T-1} a_i \log B(T+i, \chi) \right| < A(T, m),$$

where

$$A(T, m) = c(m)G(T) + H(T), \tag{3.2}$$

$$\begin{aligned} G(x) &= \frac{x^{3/2}}{U(x)} \left[ 1 + 2^{3/2} + 5\lambda + \frac{7\lambda}{\log x} + \frac{4\lambda}{(\log x)^2} \right] + \\ &+ \frac{12}{x \log x} + \frac{8}{x(\log x)^2} + \frac{4}{x(\log x)^3} + \frac{12}{x^2 \log x} + \frac{15}{2x^2(\log x)^2} + \frac{3}{x^2(\log x)^3}, \\ H(x) &= \frac{4C\lambda x^{3/2}}{U(x)} + \frac{3C}{(\log 2)x^{2/3}} + \frac{6}{x} + \frac{10+2 \log 2}{x \log x} + \frac{6}{x(\log x)^2} + \\ &+ \frac{2}{x(\log x)^3} + \frac{4}{x^2 \log x} + \frac{5}{2x^2(\log x)^2} + \frac{1}{x^2(\log x)^3}, \end{aligned}$$

and  $U(x)$ ,  $C$ ,  $\lambda$ ,  $c(m)$  have been defined above.

In our case, we have  $m = p$  and we set

$$B(T+i) = B(T+i, \chi_p), \quad S(T, p) = \sum_{i=0}^{T-1} a_i \log B(T+i).$$

Putting  $E(T, p) = \log L(1, \chi_p) - S(T, p)$ , we get

$$|E(T, p)| < A(T, p).$$

Since

$$\exp(E(T, p) + S(T, p)) = L(1, \chi_p),$$

we have by (1.2)

$$e^{E(T, p)} (\tilde{h} + \delta) = h (= h(p)),$$

where

$$\tilde{h} = \text{Ne} \left( \frac{\sqrt{p} e^{S(T, p)}}{R_2 \log 4} \right)$$

and

$$\delta = \frac{\sqrt{p} e^{S(T, p)}}{R_2 \log 4} - \tilde{h}.$$

Suppose we suspect that  $\tilde{h} + g$  (odd) is the value of  $h$ , where  $|g|$  is a small even integer (we used  $|g| \leq 4$ ). We also assume that we have an odd factor  $h_1$  ( $\geq 1$ ) of  $\tilde{h} + g$  which must also divide  $h$ . This will be explained in the next section. Assume further that  $h_1 \geq |g - \delta|/2$  and put  $h_2 = h/h_1$ . Evidently,

$$e^{E(T, p)} \left( \frac{\tilde{h} + \delta}{h_1} \right) = h_2. \quad (3.3)$$

We consider two cases.

**Case 1.**  $E(T, p) > 0$ .

In this case, we see from (3.3) that

$$h_2 > \frac{\tilde{h} + \delta}{h_1} = \frac{\tilde{h} + g}{h_1} - \frac{g - \delta}{h_1}.$$

If

$$e^{A(T, p)} < \frac{\tilde{h} + g + 2h_1}{\tilde{h} + \delta},$$

then

$$e^{E(T, p)} < \frac{\tilde{h} + g + 2h_1}{\tilde{h} + \delta}$$

and from (3.3)

$$h_2 < \frac{\tilde{h} + g}{h_1} + 2.$$

Since  $(g - \delta)/h_1 \leq 2$ , we get

$$\frac{\tilde{h} + g}{h_1} - 2 < h_2 < \frac{\tilde{h} + g}{h_1} + 2.$$

It follows that, because  $h_2$  must be odd,  $h_2 = (\tilde{h} + g)/h_1$  or  $h = \tilde{h} + g$ .

**Case 2.**  $E(T, p) < 0$ .

In this case we get

$$h_2 < \frac{\tilde{h} + g}{h_1} - \frac{g - \delta}{h_1}$$

from (3.3). Suppose  $(\tilde{h} + g)/h_1 \geq 3$ . If

$$e^{A(T, p)} < \frac{\tilde{h} + \delta}{\tilde{h} + g - 2h_1},$$

then

$$e^{-A(T,p)} > \frac{\tilde{h} + g - 2h_1}{\tilde{h} + \delta}$$

and

$$\frac{\tilde{h} + g}{h_1} - 2 < e^{-A(T,p)} \frac{\tilde{h} + \delta}{h_1} < e^{E(T,p)} \frac{\tilde{h} + \delta}{h_1} = h_2.$$

Since  $(g - \delta)/h_1 \geq -2$ , we get

$$\frac{\tilde{h} + g}{h_1} - 2 < h_2 < \frac{\tilde{h} + g}{h_1} + 2$$

and  $h = \tilde{h} + g$ . If  $(\tilde{h} + g)/h_1 < 3$ , then  $\tilde{h} + g = h_1$ . If  $h_2 \geq 3$ , then

$$e^{E(T,p)} \geq \frac{3h_1}{\tilde{h} + \delta} = \frac{3h_1}{h_1 - g + \delta} \geq 1,$$

a contradiction. Hence  $h_2 = 1$  and  $\tilde{h} + g = h$ .

Recapitulating, we have shown that if  $\tilde{h} + g \geq 3h_1$ ,  $|g - \delta| \leq 2h_1$  and

$$e^{A(T,p)} < \min \left\{ \frac{\tilde{h} + g + 2h_1}{\tilde{h} + \delta}, \frac{\tilde{h} + \delta}{\tilde{h} + g - 2h_1} \right\},$$

then  $h = \tilde{h} + g$ . Also, if  $\tilde{h} + g = h_1$ ,  $|g - \delta| \leq 2h_1$  and

$$e^{A(T,p)} < \frac{3h_1}{\tilde{h} + \delta},$$

then  $h = \tilde{h} + g$ . Thus, as long as we have some  $h_1$  such that  $2h_1 \geq |g - \delta|$  and some  $T$  such that  $\exp(A(T,p))$  is sufficiently small, we can find the value of  $h$ . As we wish to limit the amount of work to evaluate  $S(T,p)$  (that is, keep  $T$  as small as possible), it is important to be able to have the smallest possible bound on  $E(T,p)$ . Notice that while our formula for  $A(T,p)$  is rather complicated, it is easy to compute because the values of  $G(T)$  and  $H(T)$  can be easily tabulated for various values of  $T$  in advance.

In Table 1, we compare our error bound  $A(T,p)$  on  $|\log L(1, \chi_p) - S(T,p)|$  (given below (3.2)) with Bach's error bound  $A'(T,p)$  (given in (3.1) and in Table 3 of [1], with  $A$  and  $B$  taken from the third and fourth column of that table). The ratio  $A(T,p)/A'(T,p)$  varies slowly (with  $p$  and  $T$ ) near 0.78 so we conclude that our error bound is about 22% sharper than Bach's error bound.

In order to test the effect of this improvement on the efficiency of our algorithm, we compared the use of both bounds for the computation of the class numbers of the 157 987 primes  $\equiv 1 \pmod{4}$  in the interval  $[5\,000\,000, 10\,000\,000]$ . For  $T = 1000$  and  $f = 10$ , in the case of our bound, our algorithm determined the class number  $h(p) = 3$  with Step 2b (see Section 1), from  $\tilde{h} = 3$ ,  $h_1 = 1$  in 18 169 cases, because (1.4) was satisfied. For these 18 169 cases this inequality was *not* satisfied with the use of Bach's error bound  $A'(T,p)$ . Most of these cases were handled in the follow-up of Step 2b, namely where a *divisor*  $h_1$  of  $h$  is found, but this increased the CPU time. In the case of our bound, our algorithm took 115 CPU seconds while 516 cases were left undetermined (those are treated with higher values of  $T$  and  $f$ , see Section 5). In case of Bach's bound, our algorithm took 149 CPU seconds, while 3070 cases were left undetermined. We conclude that the use of our error bound  $A(T,p)$  increases the efficiency of our program with at least 20% compared with the use of Bach's error bound.

#### 4. FINDING A DIVISOR OF $h$

In this section we will explain how to find a divisor of the class number  $h$  when we have an expectation as to what  $h$  is. In order to do this, we must first derive a technique for detecting whether or not a given reduced ideal is principal.

| $p$             | $T$  | $A'(T, p)$ | $A(T, p)$ | $A(T, p)/A'(T, p)$ |
|-----------------|------|------------|-----------|--------------------|
| 9 999 999 937   | 100  | 4.5704     | 3.5820    | 0.7837             |
|                 | 500  | 1.3418     | 1.0476    | 0.7808             |
|                 | 1000 | 0.8256     | 0.6450    | 0.7813             |
|                 | 5000 | 0.2841     | 0.2224    | 0.7827             |
| 99 999 999 977  | 100  | 4.9685     | 3.8766    | 0.7802             |
|                 | 500  | 1.4596     | 1.1332    | 0.7764             |
|                 | 1000 | 0.8983     | 0.6978    | 0.7768             |
|                 | 5000 | 0.3094     | 0.2407    | 0.7779             |
| 199 999 999 949 | 100  | 5.0884     | 3.9653    | 0.7793             |
|                 | 500  | 1.4950     | 1.1590    | 0.7752             |
|                 | 1000 | 0.9201     | 0.7136    | 0.7756             |
|                 | 5000 | 0.3169     | 0.2462    | 0.7767             |

Table 1: Comparison of  $A'(T, p)$  and  $A(T, p)$ .

We define, as before,

$$\mathcal{L} = \{\mathfrak{b}_1 (= (1)), \mathfrak{b}_2, \dots, \mathfrak{b}_{m-1}\},$$

where  $\zeta_m > t + B + 1$  and  $\zeta_{m-1} \leq t + B + 1$ . Suppose  $\mathfrak{a}$  is any reduced ideal. We define

$$\mathcal{L}(\mathfrak{a}) = \{\mathfrak{a}_1 (= (\mathfrak{a})), \mathfrak{a}_2, \dots, \mathfrak{a}_{m'-1}\}$$

where  $\mathfrak{a}_{i+1}$  is obtained from  $\mathfrak{a}_i$  by the continued fraction algorithm. Here,  $\zeta'_{m'} > 2t + B + 1$ ,  $\zeta'_{m'-1} \leq 2t + B + 1$ .

**Lemma 4.1** *If  $\mathfrak{a}$  is a reduced principal ideal and  $\mathfrak{a} = (\alpha)$  with  $1 \leq \alpha < \epsilon$ , then if  $\mathfrak{b}_1 \notin \mathcal{L}(\mathfrak{a})$ , we have  $\epsilon > \alpha 2^{2t}$ .*

**Proof.** We have  $\mathfrak{a}_i = (\Psi'_i) \mathfrak{a}_1$ , where  $\mathfrak{a}_1 = \mathfrak{a} = (\alpha)$  with  $1 \leq \alpha < \epsilon$ . Since  $\mathfrak{a}$  is principal and reduced, so are all the ideals in  $\mathcal{L}(\mathfrak{a})$

$$\Rightarrow \mathfrak{a}_1 = \mathfrak{b}_k, \mathfrak{a}_2 = \mathfrak{b}_{k+1}, \dots, \mathfrak{a}_{m'-1} = \mathfrak{b}_{k+m'-2}$$

for some  $k \in \mathbb{Z}^+$ . If  $\epsilon = \Psi'_i \alpha$ , we must have  $\mathfrak{a}_i = \mathfrak{b}_1 = (\epsilon) = (1)$ . Since  $\mathfrak{b}_1 \notin \mathcal{L}(\mathfrak{a})$ , it follows that  $i > m' - 1$  and

$$\epsilon > \alpha \Psi'_{m'-1} = \alpha \Psi_{m'}/\psi'_{m'-1} \geq \alpha 2^{\zeta'_{m'}-1}/\psi'_{m'-1} > \alpha 2^{2t+B}/\psi'_{m'-1} > \alpha 2^{2t}$$

$$(\psi'_{m'-1} < 2^B). \quad \square$$

Now suppose that  $\mathfrak{a}$  is principal. Without loss of generality,  $\mathfrak{a} = (\alpha)$  ( $1 \leq \alpha < \epsilon$ ). Suppose also that  $\mathfrak{a} \notin \mathcal{L}$ . In this case we must have  $\alpha > 2^t$ . We can define  $k \in \mathbb{Z}$  ( $k \geq 2$ ) by

$$2^{(k-1)t} \leq \alpha < 2^{kt}.$$

Since

$$2^{(n-1)t} \leq \epsilon < 2^{nt}$$

we get

$$2^{(n-k-1)t} < \epsilon/\alpha < 2^{(n-k+1)t}.$$

If  $\mathfrak{b}_1 \notin \mathcal{L}(\mathfrak{a})$ , we see by Lemma 4.1 that  $\epsilon/\alpha > 2^{2t}$ ; hence

$$2t < (n - k + 1)t \Rightarrow n - k + 1 > 2 \Rightarrow k \leq n - 1.$$

**Theorem 4.2** *If  $j = \lceil \frac{k}{2} \rceil$ , then  $\mathfrak{b}(2jt) \in \mathcal{L}(\mathfrak{a})$ .*

**Proof.** Let  $\mathfrak{b}(2jt) = (\Psi_l)$ , where

$$\Psi_l \leq 2^{2jt}, \quad \Psi_{l+1} > 2^{2jt}.$$

Consider  $\Psi_l/\alpha$ . We know that  $\alpha = \Psi_q$  for some  $q$  (we are assuming that  $\mathfrak{a}$  is principal and reduced) and since

$$2^{(k-1)t} < \alpha < 2^{kt} \leq 2^{2jt},$$

we see that

$$\Psi_q \leq 2^{2jt} \Rightarrow q \leq l \Rightarrow \Psi_l/\alpha = \Psi_l/\Psi_q \geq 1.$$

Also,

$$\Psi_l/\alpha \leq 2^{2jt}/2^{(k-1)t} = 2^{(2j-k)t+t} < 2^{2t}.$$

Now  $\Psi_l = \alpha\Psi'_s$  and  $1 \leq \Psi'_s < 2^{2t}$ ; consequently,

$$\mathfrak{a}_s = (\Psi'_s)\mathfrak{a}_1 = (\Psi'_s\alpha) = (\Psi_l) = \mathfrak{b}(2jt).$$

Also, since  $1 \leq \Psi'_s < 2^{2t}$ , we have  $\mathfrak{a}_s \in \mathcal{L}(\mathfrak{a})$ . □

We note that if  $2|n$ , then  $k = n - 2$  means that  $k$  is even; thus,

$$j = \left\lceil \frac{k}{2} \right\rceil = \frac{n}{2} - 1 = i' - 1 \leq i,$$

by Corollary 2.6. If  $k < n - 2$  then  $k \leq n - 3$ ; hence,

$$\frac{k}{2} + \frac{1}{2} \leq \frac{n}{2} - 1$$

and

$$j = \left\lceil \frac{k}{2} \right\rceil \leq \frac{n}{2} - 1 \leq i.$$

If  $2 \nmid n$ , then  $k \leq n - 2 = 2i' - 1$  so we get

$$\frac{k}{2} \leq i' - \frac{1}{2} \Rightarrow j = \left\lceil \frac{k}{2} \right\rceil \leq \frac{k}{2} + \frac{1}{2} \leq i' = i,$$

by Corollary 2.6. Thus, if we put  $\mathcal{B} = \{\mathfrak{b}_1 (= \mathfrak{b}(0)), \mathfrak{b}(2t), \mathfrak{b}(4t), \dots, \mathfrak{b}(2it)\}$ , we have the following theorem.

**Theorem 4.3** *If  $\mathfrak{a}$  is any reduced ideal and  $\mathfrak{a} \notin \mathcal{L}$ , then  $\mathfrak{a}$  is principal if and only if*

$$\mathcal{B} \cap \mathcal{L}(\mathfrak{a}) \neq \emptyset.$$

**Proof.** Certainly, if  $\mathfrak{a}$  is not principal, then  $\mathcal{B} \cap \mathcal{L}(\mathfrak{a}) = \emptyset$ . If  $\mathfrak{a}$  is principal and  $\mathfrak{a} \notin \mathcal{L}$ , we have seen already that  $\mathfrak{b}(2jt) \in \mathcal{L}(\mathfrak{a})$  for some  $j$  such that  $0 \leq j \leq i$ . Hence,  $\mathcal{B} \cap \mathcal{L}(\mathfrak{a}) \neq \emptyset$ . □

We now have our algorithm for principality testing.

**Algorithm 4.4** (Determine whether or not a given reduced ideal  $\mathfrak{a}$  is principal.)

1. If  $\mathfrak{a} \in \mathcal{L}$ , then  $\mathfrak{a}$  is principal and the algorithm terminates.
2. Compute  $\mathfrak{a}_1, \mathfrak{a}_2, \dots$  and check whether  $\mathfrak{a}_q \in \mathcal{B}$  ( $q = 1, 2, \dots, m' - 1$ ). (Note that when we need to execute this algorithm, we usually have  $R_2 < M/P$ ; hence  $\mathcal{B}$  has been computed previously in our

Algorithm 2.7.

3. If  $\mathfrak{a}_q \in \mathcal{B}$ , then  $\mathfrak{a}$  is principal. If  $\mathcal{B} \cap \mathcal{L}(\mathfrak{a}) = \emptyset$ , then  $\mathfrak{a}$  is nonprincipal.

Suppose  $q$  is a prime and  $q^\alpha \parallel \tilde{h} + g$ . We can produce an algorithm which often determines a nontrivial divisor of  $h$ .

**Algorithm 4.5** (Determine that  $\tilde{h} + g \neq h$  or find a nontrivial divisor of  $h$ .)

1. Select a new ideal  $\mathfrak{s}$  from a stock  $\mathcal{S}$  (to be described later) of reduced ideals.
2. Test if  $\mathfrak{s}$  is principal. If so, return to Step 1. If a reduced ideal  $\mathfrak{t}$  equivalent to  $\mathfrak{s}^{\tilde{h}+g}$  is not principal, we know that  $h \neq \tilde{h} + g$  and we terminate the algorithm. (Of course, if  $\mathfrak{s}^{\tilde{h}+g}$  is principal, this causes us to suspect even more that  $h = \tilde{h} + g$ .)
3. If  $\mathfrak{s}^{(\tilde{h}+g)/q^\alpha}$  is principal, go back to Step 1.
4. Compute the least value of  $\beta (> 0)$  such that  $\mathfrak{s}^{(\tilde{h}+g)/q^\beta}$  is not principal. Then  $q^{\alpha-\beta+1}$  is a nontrivial divisor of  $h$ .

**Proof** (of correctness). Clearly, if  $\mathfrak{t}$  is not principal, then  $h \neq \tilde{h} + g$ . If  $\mathfrak{s}^{\tilde{h}+g}$  is principal, we let  $\omega$  be the least positive integer such that  $\mathfrak{s}^\omega$  is principal ( $\omega > 1$ ). We know that since  $\mathfrak{s}^h$  is principal, we must have  $\omega|h$ . Now  $\omega|(\tilde{h} + g)/q^{\beta-1}$  and  $\omega \nmid (\tilde{h} + g)/q^\beta$ . Hence,  $q^\gamma \parallel \omega$ , where  $q^\gamma \parallel (\tilde{h} + g)/q^{\beta-1}$ . Since  $\gamma \geq \alpha - \beta + 1$  and  $\alpha \geq \beta$ , we have proved the correctness of Algorithm 4.5.  $\square$

The ideals in the stock  $\mathcal{S}$  can be easily developed from a table of small odd primes  $\mathcal{R} = \{r_1, r_2, \dots, r_n\}$ ,  $r_1 = 3, r_2 = 5, \dots$  (In the computations described in Section 5, we used  $n = 34$ .) For each  $r \in \mathcal{R}$  the table should contain a list of all the quadratic residues  $a$  of  $r$  and the odd square root  $x$  of  $a \pmod r$  which is between 0 and  $r$ . To create  $\mathcal{S}$  for a given  $p$ , we need only find the value of  $r$  such that  $p \equiv a \pmod r$ . Then  $\mathfrak{s} = \left[ r, \frac{x + \sqrt{p}}{2} \right]$  is an ideal of  $\mathbb{Q}(\sqrt{p})$  and since  $r < \sqrt{p}/2$ ,  $\mathfrak{s}$  is reduced already. Although, in principle, Algorithm 4.5 might not find a divisor of  $h$  (this would certainly be the case if  $\tilde{h} + g \neq h$ ), in practice we found that it worked very well. Thus, if we know the primes that divide  $\tilde{h} + g$ , we can often find a nontrivial divisor  $h_1$  of  $h$ . If we are unsuccessful in this effort, we change the value of  $g$  and try again. If this fails for all even  $|g| \leq 4$ , we put the prime  $p$  into a special set of primes  $\mathcal{P}$  and deal with them separately.

## 5. IMPLEMENTATION AND COMPUTATIONAL RESULTS

### 5.1 Implementation

We implemented our algorithm for computing  $h(p)$  for primes  $p \equiv 1 \pmod 4$  in Fortran 77 and we tested and ran it on one processor of CWI's SGI Origin 2000 computer system.<sup>1</sup> Here, we describe the six different steps.

*Step 1a* (find an integral multiple  $M$  of  $R_2$ )

This step is fully described in [9]. First, an approximation  $S(T, p)$  of  $L(1, \chi_p)$  is computed, for suitable  $T$ , and then an approximation of a multiple of  $R_2$  using the analytic class number formula (1.2). Next, with Algorithm 5.4 of [9], an integral multiple  $M$  of  $R_2$  is computed from this approximation.

*Step 1b* (compute  $R_2$  from  $M$  or prove that  $R_2 > M/P$ , where  $P$  is some small prime)

This step is carried out with help of Algorithm 2.7 as given in Section 2, for suitable  $f$ . Some experiments revealed that  $P = 11$  was sufficient for our purpose.

*Step 1c* (given that  $R_2 > M/11$ , find  $R_2$ )

<sup>1</sup>This system consists of 16 R10000/250 MHz processors and 16 R12000/300 MHz processors. In our runs, we did not distinguish between the two types of processors so fluctuations of about 20% in CPU times in comparable jobs were accepted as being caused by the two different types of processors.



This step is carried out with help of Algorithm 2.8 as given in Section 2.

*Step 2a* (compute an approximation  $\tilde{h}$  of  $h$ )

This is done with help of the approximation  $S(T, p)$  of  $\log L(1, \chi_p)$  as computed in Step 1a, and the class number formula (1.2). We take  $\tilde{h}$  to be the nearest odd integer to  $\sqrt{p} \exp(S(T, p)) / (R_2 \log 4)$  and  $\delta$  to be the difference  $\sqrt{p} \exp(S(T, p)) / (R_2 \log 4) - \tilde{h}$ , with  $|\delta| \leq 1$ .

*Step 2b* (try to compute  $h$  from  $\tilde{h}$ )

This is the crucial step in our algorithm. We start to carry out this step, as described in Section 1, with  $g = 0$ . If this does not lead to the conclusion that  $h = \tilde{h} + g$ , we repeat Step 2b with  $g = 2$ . The next tries, as long as we do not find the value of  $h$ , are done for, successively,  $g = -2$ ,  $g = 4$ , and  $g = -4$ . If unsuccessful at this stage, we turn to Step 2c.

In Step 2b, an odd divisor  $h_1 > 1$  of  $\tilde{h} + g$  has to be found. This is done with the help of Algorithm 4.5, described in Section 4. This, in turn, needs to test whether a given reduced ideal is principal. Algorithm 4.5, described in Section 4, does this job.

*Step 2c* (treat the remaining primes)

For these ‘stubborn’ cases, we resort to the PARI-GP package, namely, the function `quadclassunit`. This is much slower than our algorithm, but the number of primes left to be treated here is so small compared with those for which our algorithm could compute the class number, that the total CPU time needed for Step 2c remains small compared with the CPU time needed for our algorithm.

## 5.2 Results

We computed  $h(p)$  for all the primes  $p \equiv 1 \pmod{4}$  below the bound  $2 \times 10^{11}$ . We made 200 runs, each covering an interval of length  $10^9$ . In each run we first applied our algorithm with  $T = 3000$ ,  $f = 3$ . For the 200 intervals which we checked, this was always successful for more than 99% of the primes and consumed a corresponding portion of the total CPU time for this run. For the remaining primes, we repeated our algorithm nine times with increasing values of  $T$  and  $f$ , namely with  $T = 3000 + j \times 500$ ,  $f = 3 + 5 \times j$ , for  $j = 1, 2, \dots, 9$ . This further decreased the number of primes for which our algorithm could *not* compute the class number. For example, the interval  $[199 \times 10^9, 200 \times 10^9]$  contains 19 217 740 primes which are  $\equiv 1 \pmod{4}$ . The numbers of primes left after each of the above ten steps was: 99309, 35016, 31396, 28690, 25193, 23366, 21808, 20566, 16060, and 3677, respectively. The CPU times for these ten steps were: 63663, 590, 319, 362, 410, 415, 441, 468, 483, and 1116 seconds, respectively. The 3677 primes left after the tenth step were treated with the PARI-GP package and this required 2650 CPU seconds.

The total CPU-time per run varied between 10 CPU hours for the 25 423 491 primes which are  $\equiv 1 \pmod{4}$  in the interval  $[1, 10^9]$  and 20 CPU hours for the 19 217 740 primes which are  $\equiv 1 \pmod{4}$  in the interval  $[199 \times 10^9, 200 \times 10^9]$ . Total CPU time was about 3000 CPU hours. Usually, we executed four runs in parallel on four processors of CWI’s Origin 2000 system. The number of primes treated in Step 2c with the PARI-GP function `quadclassunit` was about 2000 for the (first) interval  $[1, 10^9]$  and about 3700 for the (last) interval  $[199 \times 10^9, 200 \times 10^9]$ . The CPU times for these primes varied between 500 and 2700 CPU seconds. Total CPU time with PARI-GP for Step 2c was about 120 CPU hours. For the last interval  $[199 \times 10^9, 200 \times 10^9]$  the average CPU time per prime for the primes treated in Steps 1a–2b was 3.5 msec. and the average CPU time per prime treated in Step 2c (with PARI-GP) was 0.72 sec. (slower by a factor of about 200).

Let

$$\pi_{4,1}(x) = \#\{p \leq x \mid p \equiv 1 \pmod{4}, p \text{ prime}\}$$

and

$$\pi_{4,1,n}(x) = \#\{p \leq x \mid p \equiv 1 \pmod{4}, p \text{ prime}, h(p) = n\}.$$

For the class numbers  $h(p) \leq 29$ , in Table 2 we compare their frequencies of occurrence with those

| $x$                | $\pi_{4,1}(x)$ | $r_1(x)$ | $r_3(x)$ | $r_5(x)$ | $r_7(x)$ | $r_9(x)$ | $r_{11}(x)$ |
|--------------------|----------------|----------|----------|----------|----------|----------|-------------|
| $10^9$             | 25423491       | 1.00976  | 0.95830  | 1.00239  | 1.00646  | 0.93604  | 1.00508     |
| $2 \times 10^9$    | 49109660       | 1.00865  | 0.96285  | 1.00125  | 1.00561  | 0.94171  | 1.00521     |
| $5 \times 10^9$    | 117474981      | 1.00739  | 0.96765  | 1.00110  | 1.00501  | 0.94989  | 1.00530     |
| $10^{10}$          | 227523275      | 1.00654  | 0.97103  | 1.00108  | 1.00426  | 0.95473  | 1.00472     |
| $2 \times 10^{10}$ | 441101890      | 1.00578  | 0.97417  | 1.00128  | 1.00371  | 0.95981  | 1.00415     |
| $5 \times 10^{10}$ | 1059822165     | 1.00494  | 0.97768  | 1.00128  | 1.00317  | 0.96569  | 1.00363     |
| $10^{11}$          | 2059020280     | 1.00437  | 0.98001  | 1.00139  | 1.00319  | 0.96930  | 1.00303     |
| $2 \times 10^{11}$ | 4003548492     | 1.00387  | 0.98214  | 1.00143  | 1.00289  | 0.97265  | 1.00306     |

| $r_{13}(x)$ | $r_{15}(x)$ | $r_{17}(x)$ | $r_{19}(x)$ | $r_{21}(x)$ | $r_{23}(x)$ | $r_{25}(x)$ | $r_{27}(x)$ | $r_{29}(x)$ |
|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| 1.00583     | 0.95228     | 1.00483     | 1.01174     | 0.95320     | 1.00873     | 0.99246     | 0.92706     | 1.01402     |
| 1.00835     | 0.95546     | 1.00647     | 1.01194     | 0.95647     | 1.00717     | 0.99228     | 0.93598     | 1.01220     |
| 1.00554     | 0.96120     | 1.00602     | 1.00765     | 0.96160     | 1.00750     | 0.99597     | 0.94677     | 1.01042     |
| 1.00515     | 0.96590     | 1.00650     | 1.00676     | 0.96732     | 1.00639     | 0.99816     | 0.95184     | 1.01074     |
| 1.00503     | 0.96923     | 1.00535     | 1.00444     | 0.97047     | 1.00575     | 0.99828     | 0.95707     | 1.00800     |
| 1.00420     | 0.97349     | 1.00465     | 1.00396     | 0.97573     | 1.00488     | 0.99909     | 0.96238     | 1.00642     |
| 1.00411     | 0.97681     | 1.00434     | 1.00410     | 0.97814     | 1.00506     | 0.99937     | 0.96578     | 1.00434     |
| 1.00362     | 0.97972     | 1.00368     | 1.00382     | 0.98074     | 1.00403     | 1.00019     | 0.96932     | 1.00348     |

Table 2: Comparison of class number frequencies with the Cohen-Lenstra heuristics.

‘predicted’ by the Cohen-Lenstra heuristics, namely, by listing the values of

$$\pi_{4,1}(x) \quad \text{and} \quad r_h(x) := \frac{\pi_{4,1,h}(x)}{\pi_{4,1}(x)} \bigg/ \frac{Cw(h)}{h},$$

for various choices of  $x$ . The ratios  $r_h(x)$  seem to tend to 1 with growing  $x$ , so Table 2 provides numerical support for the Cohen-Lenstra heuristics. Notice that for  $h = 1$  and for  $h$  prime, the frequencies  $\pi_{4,1,h}(x)/\pi_{4,1}(x)$  seem to tend to their Cohen-Lenstra limit  $Cw(h)/h$  from above, whereas for composite values of  $h$ , this pattern is reversed. Moreover, the *speed* of convergence is higher for  $h = 1$  and  $h$  prime than for composite  $h$ .

Together with the class numbers, we computed the function  $H(x)$ . Table 3 tabulates  $H(x)$  for various values of  $x$ , together with  $8 \times H(x)/x$ , which should tend to 1, according to Hooley’s conjecture. Figure 1 plots the function  $8 \times H(x)/x$  for  $x = i \times 10^9, i = 1, 2, \dots, 200$ . The scattered points show the ‘local contributions’ to this function, namely the values

$$8 \frac{H(i * 10^9) - H((i-1) * 10^9)}{10^9}, \quad \text{for } i = 1, \dots, 200.$$

Table 3 and Figure 1 confirm that the function  $8 \times H(x)/x$  increases on the interval where we have computed it. The majority of the local contributions lie *above* the ‘average’  $8 \times H(x)/x$ , and Figure 1 does not give any clue that this ‘behaviour’ would change after our bound  $2 \times 10^{11}$ . Figure 1 also illustrates that if the function  $8 \times H(x)/x$  converges to 1, it converges extremely slowly.

### 5.3 Examples

**Example 1** We take  $p = 97843343893$  as in [9] with  $T = 1000$  and  $f = 10$ . Step 1a finds  $S(T, p) = 0.3765342$  and  $M = 329944.5389420387$  for the integral multiple of  $R_2$ .<sup>2</sup>

<sup>2</sup>The value of  $kR_2$  reported in [9] is three times the value given here, because of a mistake HtR made in [9] in the programming of the Kronecker symbol. This is explained and corrected in [10]. The consequence of this mistake is that

| $x$                | $H(x)$      | $8 \times H(x)/x$ |
|--------------------|-------------|-------------------|
| $10^9$             | 101284007   | 0.81027           |
| $2 \times 10^9$    | 203601670   | 0.81441           |
| $5 \times 10^9$    | 511808671   | 0.81889           |
| $10^{10}$          | 1027420829  | 0.82194           |
| $2 \times 10^{10}$ | 2062604790  | 0.82504           |
| $5 \times 10^{10}$ | 5175931981  | 0.82815           |
| $10^{11}$          | 10386588068 | 0.83093           |
| $2 \times 10^{11}$ | 20841205517 | 0.83365           |

Table 3: Some values of  $H(x)$  and  $8 \times H(x)/x$ .

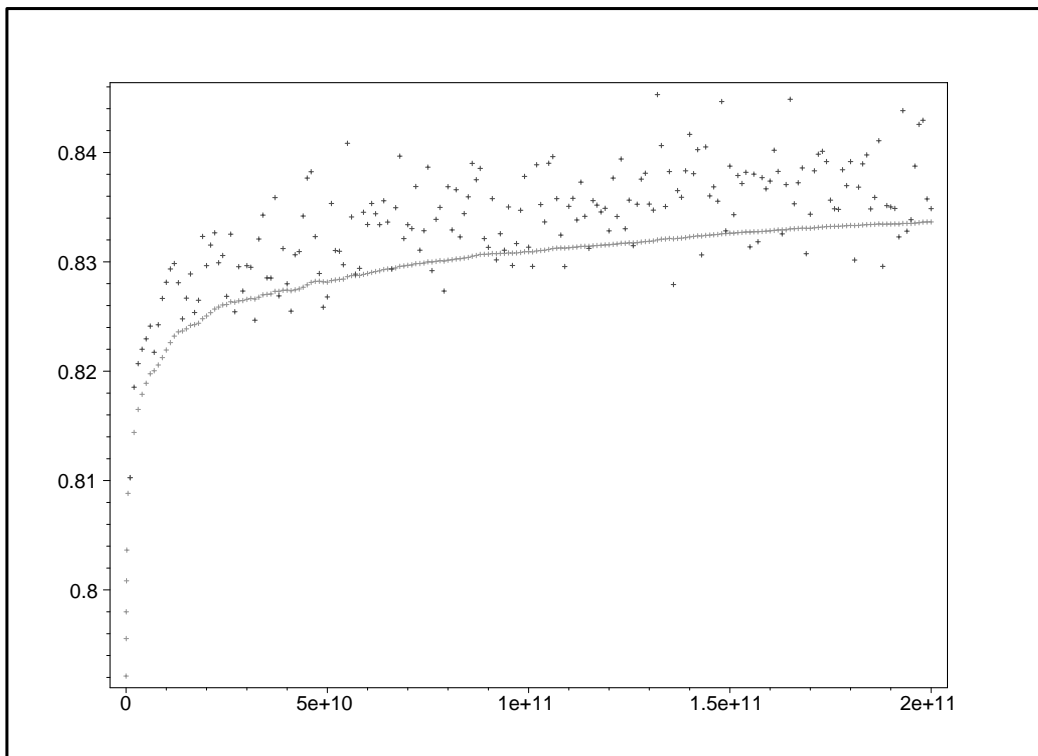


Figure 1: Plot of  $8 \times H(x)/x$  and its local contributions for  $x = i \times 10^9, i = 1, 2, \dots, 200$

In Step 1b, Algorithm 2.7 is carried out, i.e., first the list  $\mathcal{L}$  is computed. In step 2 of Algorithm 2.7, we did not find a match of  $\mathfrak{b}(2it)$  neither of  $\bar{\mathfrak{b}}(2it)$  with some element of  $\mathcal{L}$ , for  $i = 1, 2, \dots, \lceil (K + 2B + 1)/2t \rceil$ , so this shows that  $R_2 > K$  with  $K = M/11 = 29994.9580856399$ .

In Step 1c, Algorithm 2.8 is carried out, i.e., it is verified that  $\mathfrak{b}(M/p) \neq \mathfrak{b}_1$ , for  $p = 3, 5, 7$ . It follows that

$$R_2 = R/\log(2) = M = 329944.5389420387.$$

In Step 2a, we compute  $\sqrt{p} \exp(S(T, p))/(R_2 \log 4) = 0.9965428$ , so that  $\tilde{h} = 1$  and  $\delta = -0.0034572$ . In Step 2b, with  $g = 0$ , for the function  $A(T, m)$  described in Section 3, we find that  $A(1000, p) = 0.6972602$ , so that  $\exp(A(1000, p)) = 2.008243$ . With  $h_1 = 1$ , we have  $3h_1/(\tilde{h} + \delta) = 3.010407$  so that  $\exp(A(T, p)) < 3h_1/(\tilde{h} + \delta)$  and we conclude that  $h(97\,843\,343\,893) = \tilde{h} = 1$ .

**Example 2** We take  $p = 990\,000\,388\,129$  with  $T = 1000$  and  $f = 10$ .

Step 1a finds  $S(T, p) = 1.895771$  and  $M = 4729385.900492189$ .

In Step 1b, Algorithm 2.7 computes the list  $\mathcal{L}$ , and no match is found of  $\mathfrak{b}(2it)$  nor of  $\bar{\mathfrak{b}}(2it)$  with some element in this list, for  $i = 1, 2, \dots, \lceil (K + 2B + 1)/2t \rceil$ , so this shows that  $R_2 > M/11$ .

In Step 1c it is verified that  $\mathfrak{b}(M/p) \neq \mathfrak{b}_1$  for  $p = 3, 7$ , but  $\mathfrak{b}(M/5) = \mathfrak{b}_1$  and  $\mathfrak{b}(M/25) \neq \mathfrak{b}_1$ . It follows that

$$R_2 = R/\log(2) = M/5 = 945877.1800984377.$$

Step 2a computes  $\sqrt{p} \exp(S(T, p))/(R_2 \log 4) = 5.0518490$ , so that  $\tilde{h} = 5$  and  $\delta = 0.0518490$ .

In Step 2b, with  $g = 0$ , we find  $\exp(A(1000, p)) = 2.117520$ . For  $h_1 = 1$ ,  $\tilde{h} + g \geq 3h_1$  and

$$\min \left\{ \frac{\tilde{h} + g + 2h_1}{\tilde{h} + \delta}, \frac{\tilde{h} + \delta}{\tilde{h} + g - 2h_1} \right\} = 1.385631,$$

so no conclusion for  $h$  is possible and we try to find a divisor of  $h$  with Algorithm 4.5. We try the divisor  $q = 5$  of  $\tilde{h} + g$  (of course). For the first ideal  $\mathfrak{s} = [6/2, (1 + \sqrt{p})/2]$  from the stock  $\mathcal{S}$ , Algorithm 4.4 finds that it is not principal. Step 2 of Algorithm 4.5 now finds a reduced ideal  $\mathfrak{t} = [486/2, (61 + \sqrt{p})/2]$  which is equivalent to  $\mathfrak{s}^{\tilde{h}+g} = \mathfrak{s}^5$ . This ideal  $\mathfrak{t}$  is found to be principal with help of Algorithm 4.4. For  $\beta = 1$ ,  $\mathfrak{s}^{(\tilde{h}+g)/q^\beta} = \mathfrak{s}$  is not principal, as we already know, and we conclude that  $q^{\alpha-\beta+1} = 5$  is a nontrivial divisor of  $h$ .

Now we repeat Step 2b with  $h_1 = 5$  (and still  $g = 0$ ). We have  $\tilde{h} + g = h_1 = 5$  and  $3h_1/(\tilde{h} + \delta) = 2.969210$ , so that  $\exp(A(1000, p)) < 3h_1/(\tilde{h} + \delta)$  and we conclude that  $h(990\,000\,388\,129) = \tilde{h} = 5$ .

**Example 3**  $p = 199\,999\,913\,213$ , the largest prime  $< 2 \times 10^{11}$  for which our algorithm could compute the class number, with  $T = 7500$ ,  $f = 48$ .

Step 1a finds  $S(T, p) = -0.4557187$  and  $M = 211269.9174290152$ .

In Step 1b, Algorithm 2.7 then finds that

$$R_2 = R/\log(2) = 454.3439084494522.$$

In Step 2a, we compute  $\sqrt{p} \exp(S(T, p))/(R_2 \log 4) = 450.1514159$ , so that  $\tilde{h} = 451$  and  $\delta = -0.80966325$ .

In Step 2b, with  $g = 0$ , we find  $\exp(A(7500, p)) = 1.209404$ . For  $h_1 = 1$ ,  $\tilde{h} + g \geq 3h_1$  and

$$\min \left\{ \frac{\tilde{h} + g + 2h_1}{\tilde{h} + \delta}, \frac{\tilde{h} + \delta}{\tilde{h} + g - 2h_1} \right\} = 1.002564,$$

---

for all the primes which are  $\equiv 5 \pmod{8}$ , our computed value of  $kR_2$  in [9] is too large by a factor of 3. Fortunately, this does not affect the result of [9], namely that the Ankeny-Artin-Chowla conjecture is true for all the primes  $p \equiv 1 \pmod{4}$  below  $10^9$  since for the verification of this conjecture any multiple of  $R_2$  will suffice, as long as this does not exceed  $8p$ .

so no conclusion for  $h$  is possible. Therefore, we try to find a divisor of  $h$  with Algorithm 4.5. We start with  $q = 11$ , the smallest prime divisor of  $\tilde{h} + g = 451$ . For the first ideal  $\mathfrak{s} = [14/2, (3 + \sqrt{p})/2]$  in the stock  $\mathcal{S}$ , Algorithm 4.4 finds that it is not principal, so Step 2 of Algorithm 4.5 now finds a reduced ideal  $\mathfrak{t} = [11738/2, (439771 + \sqrt{p})/2]$  which is equivalent to  $\mathfrak{s}^{\tilde{h}+g} = \mathfrak{s}^{451}$ . With help of Algorithm 4.4, this ideal is found to be nonprincipal, so we conclude that  $h \neq \tilde{h} + g$ .

Step 2b is repeated now with  $g = -2$  so  $\tilde{h} + g = 449$ . With  $h_1 = 1$ , (1.4) is not satisfied, so no conclusion for  $h$  can be drawn. Therefore, we try to find a divisor of  $h$ . Since 449 is prime, we try  $q = 449$  in Algorithm 4.5. For the first ideal  $\mathfrak{s} = [14/2, (3 + \sqrt{p})/2]$  in the stock  $\mathcal{S}$ , Algorithm 4.4 finds that it is not principal, so Step 2 of Algorithm 4.5 now finds a reduced ideal  $\mathfrak{t} = [380938/2, (367115 + \sqrt{p})/2]$  which is equivalent to  $\mathfrak{s}^{\tilde{h}+g} = \mathfrak{s}^{449}$ . With help of Algorithm 4.4, this ideal is found to be principal. For  $\beta = 1$ ,  $\mathfrak{s}^{(\tilde{h}+g)/q^\beta} = \mathfrak{s}$  is not principal, as we already know, and we may conclude that  $q^{\alpha-\beta+1} = 449$  is a nontrivial divisor of  $h$ .

Now we repeat Step 2b with  $h_1 = 449$  (and still  $g = -2$ ). We have  $\tilde{h} + g = h_1 = 449$  and  $3h_1/(\tilde{h} + \delta) = 2.992068$ , so that  $\exp(A(7500, p)) < 3h_1/(\tilde{h} + \delta)$  and we conclude that  $h(199\,999\,913\,213) = \tilde{h} + g = 449$ .

**Example 4**  $p = 199\,999\,649\,533$  (the largest prime  $< 2 \times 10^{11}$  for which our algorithm could not compute the class number) with  $T = 7500$ ,  $f = 48$ .

Step 1a finds  $S(T, p) = -0.3602558$  and  $M = 228674.1622363300$ .

In Step 1b, Algorithm 2.7 then finds that

$$R_2 = R/\log(2) = 47.12987680055535.$$

In Step 2a, we compute  $\sqrt{p} \exp(S(T, p))/(R_2 \log 4) = 4774.2565225$ , so that  $\tilde{h} = 4775$  and  $\delta = -0.74347755$ .

In Step 2b, with  $g = 0$ , we find  $\exp(A(7500, p)) = 1.209404$ . For  $h_1 = 1$ ,  $\tilde{h} + g \geq 3h_1$  and

$$\min \left\{ \frac{\tilde{h} + g + 2h_1}{\tilde{h} + \delta}, \frac{\tilde{h} + \delta}{\tilde{h} + g - 2h_1} \right\} = 1.000263,$$

so no conclusion for  $h$  is possible. Therefore, we try to find a divisor of  $h$  with Algorithm 4.5. We start with  $q = 5$ , the smallest prime divisor of  $\tilde{h} + g = 4775$ . For the first ideal  $\mathfrak{s} = [6/2, (1 + \sqrt{p})/2]$  in the stock  $\mathcal{S}$ , Algorithm 4.4 finds that it is not principal, so Step 2 of Algorithm 4.5 now finds a reduced ideal  $\mathfrak{t} = [60238/2, (430595 + \sqrt{p})/2]$  which is equivalent to  $\mathfrak{s}^{\tilde{h}+g} = \mathfrak{s}^{4775}$ . With help of Algorithm 4.4, this ideal is found to be nonprincipal, so we conclude that  $h \neq \tilde{h} + g$ .

Step 2b is repeated now with, successively,  $g = -2, 2, -4, 4$ , but similarly as for  $g = 0$ , this leads to the conclusion that  $h \neq 4773, 4777, 4771, 4779$ .

Step 2c now resorts to PARI-GP's function `quadclassunit` which returns  $h(199\,999\,649\,533) = 4785$ .

## References

1. E. Bach, Improved approximations for Euler products, *Number Theory, CMS Conference Proceedings*, vol. 15, Amer. Math. Soc., Providence, RI, 1995, 13–28.
2. H. Cohen and H.W. Lenstra, Jr., Heuristics on class groups, *Number Theory, Lecture Notes in Mathematics 1052*, Springer, Berlin, 1984, 26–36.
3. H. Cohen and H.W. Lenstra, Jr., Heuristics on class groups of number fields, *Number Theory, Lecture Notes in Mathematics 1068*, Springer, Berlin, 1984, pp. 33–62.
4. C. Hooley, On the Pellian equation and the class number of indefinite binary quadratic forms, *J. reine angew. Math.*, 353(1984), 98–131.
5. M.J. Jacobson, Jr., Experimental results on class groups of real quadratic fields (extended abstract), *Algorithmic Number Theory - ANTS-III* (Portland, Oregon), Lecture Notes in Computer Science 1423, Springer-Verlag, Berlin, 1998, pp. 463–474.
6. M.J. Jacobson, Jr., R.F. Lukes and H.C. Williams, An investigation of bounds for the regulator of quadratic fields, *Experimental Math.* 4(1995), 211–225.
7. M.J. Jacobson, Jr. and H.C. Williams, New quadratic polynomials with high densities of prime values, *Math. Comp.*, to appear.
8. H.W. Lenstra, Jr., On the calculation of regulators and class numbers of quadratic fields, *London Math. Soc. Lecture Note Series*, 56(1982), 123–150.
9. A.J. van der Poorten, H. te Riele and H.C. Williams, Computer verification of the Ankeny-Artin-Chowla conjecture for all primes less than 100000000000, *Math. Comp.*, 70(2001), 1311–1328.
10. H. te Riele and H.C. Williams, Corrigenda and addition to: “Computer verification of the Ankeny-Artin-Chowla conjecture for all primes less than 100 000 000 000”, to appear in *Math. Comp.*
11. A.J. Stephens and H.C. Williams, Computation of real quadratic fields with class number one, *Math. Comp.* 51(1988), 809–824.