MATHEMATICS OF COMPUTATION
VOLUME 54, NUMBER 190
APRIL 1990, PAGES 855–868

p-ADIC COMPUTATION OF REAL QUADRATIC CLASS NUMBERS

J. BUCHMANN, J. W. SANDS, AND H. C. WILLIAMS

ABSTRACT. Let $\mathcal{K}$ be any real quadratic field and let $h_{\mathcal{K}}$ be the class number of $\mathcal{K}$. A method utilizing the p-adic class number formula for $\mathcal{K}$ is described for evaluating $h_{\mathcal{K}}$. The technique was programmed for a micro VAX II computer and run on all fields $\mathcal{K}$ with radicand $< 10^6$.

1. INTRODUCTION

Let $\mathcal{K}$ be any real quadratic field with discriminant d. In 1965 Slavutskii [5] advocated the use of p-adic methods for determining the class number $h_{\mathcal{K}}$ of $\mathcal{K}$. He first showed that $h_{\mathcal{K}} < \sqrt{d}$ and then exhibited a formula which could be used to determine $h_{\mathcal{K}} \pmod{p^l}$ for any prime $p \nmid d$. Thus, if $p^l > \sqrt{d}$ and we have determined by this formula the value of $h_{\mathcal{K}}$ modulo p^l, then the exact value of $h_{\mathcal{K}}$ is easily deduced. The purpose of this paper is to discuss a large-scale computer implementation of a much modified version of this idea. We determine a somewhat better bound on $h_{\mathcal{K}}$, a simpler version of Slavutskii's p-adic formula for finding $h_{\mathcal{K}} \pmod{p}$ when $p \nmid d$, and describe an efficient computer algorithm for finding $h_{\mathcal{K}}$. While this technique works reasonably well for smaller values of d, the main difficulty is that the method is of time complexity $O(d^{1+\varepsilon})$ for any $\varepsilon > 0$. Thus, for larger values of d the method is much less efficient than other available procedures for determining $h_{\mathcal{K}}$ (see, for example, the survey paper of Mollin and Williams [4]).

We first set $D = d$ when $d \equiv 1 \pmod 4$ and $D = d/4$ otherwise. That is, D is the (square-free) radicand of $\mathcal{K}$. Now

$$h_{\mathcal{K}} = \frac{\sqrt{d}}{2 \log \varepsilon_1} L(1, \chi), \tag{1.1}$$

where ε_1 is the fundamental unit of $\mathcal{K}$ and $L(1, \chi)$ is the Dirichlet series given by

$$L(1, \chi) = \sum_{n=1}^{\infty} \chi(n)/n,$$

Received January 18, 1989.
1980 *Mathematics Subject Classification* (1985 *Revision*). Primary 11R11, 11Y40, 11R29.
The second author's research was supported by NSF Vermont EPSCoR Grant RII-8610679.
The third author's research was supported by NSERC of Canada Grant A7649.

where $\chi(n) = (d/n)$ with (d/n) being the Kronecker symbol. If we refer to the upper bounds on $L(1, \chi)$ given in Stanton, Sudler, and Williams [6], we see that for $D > 100$ we get

$$L(1, \chi) < \begin{cases} (\log D)/4 + 1.3 & (d = 4D), \\ (\log D)/2 + .33 & (d = D \equiv 1 \pmod 8), \\ (\log D)/6 + .87 & (d = D \equiv 5 \pmod 8). \end{cases} \tag{1.2}$$

By using (1.1) and (1.2) we can prove

Theorem 1.1. *If $\mathcal{K}$ is any real quadratic field with class number $h_{\mathcal{K}}$ and radicand D, then*

$$h_{\mathcal{K}} < \sqrt{D}/2 + 2\sqrt{D}/\log D.$$

Proof. If $D < 100$, we can verify the theorem directly. Suppose $D > 100$. If $d = 4D$, then $\varepsilon_1 = r + s\sqrt{D}$ $(r, s \in \mathbf{Z})$ and $r^2 - s^2 D = \pm 1$; hence,

$$\varepsilon_1 \geq \sqrt{D-1} + \sqrt{D} = \sqrt{D}(1 + \sqrt{1 - D^{-1}});$$

since $\sqrt{1 - D^{-1}} > 1 - (2D-1)^{-1}$, it follows that

$$\log \varepsilon_1 > \log \sqrt{D} + \log(2 - (2D-1)^{-1}).$$

By (1.1) and the first case of (1.2) we have

$$h_{\mathcal{K}} < \sqrt{D}\left(\frac{(\log D)/4 + 1.3}{(\log D)/2 + \log(2 - (2D-1)^{-1})}\right) < \sqrt{D}\left(\frac{1}{2} + \frac{2}{\log D}\right).$$

By noting that $\varepsilon_1 = r + s\sqrt{D}$ $(r, s \in \mathbf{Z})$ when $d \equiv 1 \pmod 8$, and $\varepsilon_1 = (r + s\sqrt{D})/2$ $(r, s \in \mathbf{Z})$ when $d \equiv 5 \pmod 8$, the result can be proved in these cases by an argument similar to the above. □

Now $h_{\mathcal{K}}$ is odd if and only if D is a prime, or $D = 2q$, or $D = qr$, where q, r are primes such that $q \equiv r \equiv -1 \pmod 4$. Thus, if we want to know the value of h, we need only determine the parity of it by factoring D and then find the value of $h \pmod p$ for some odd prime $p > \sqrt{D}/4 + \sqrt{D}/\log D$. Given these results, it is a simple matter to find a positive integer congruent to $h_{\mathcal{K}} \pmod{2p}$, and since $2p > \sqrt{D}/2 + \sqrt{D}/(2\log D) > h_{\mathcal{K}}$, this must be the value of $h_{\mathcal{K}}$. In the next sections we will discuss how to find the value of $h_{\mathcal{K}} \pmod p$.

2. The p-adic class number formula

In preparation for applications, we state the p-adic class number formula of Leopoldt [3] and derive the relevant consequences. Our treatment follows Washington [9], where the proof and other references can be found. The details of obtaining the correct sign are due to Amice and Fresnel [1].

Let $\mathcal{K}$ be any totally real abelian extension of the rational number field $\mathbf{Q}$, $\mathbf{C} \supset \mathcal{K}$, and let χ run through the corresponding group of primitive even Dirichlet characters. Put $n = [\mathcal{K} : \mathbf{Q}]$ and d equal to the discriminant of $\mathcal{K}$.

Choose an integral basis $\omega_1, \ldots, \omega_n$ for the ring of integers $\mathbf{O}_{\mathscr{K}}$ of $\mathscr{K}$ and fundamental units $\varepsilon_1, \ldots, \varepsilon_{n-1}$ of $\mathbf{O}_{\mathscr{K}}$ so that $R(\mathscr{K}) = \det_{i,j}(\log|\varepsilon_i^{(j)}|) > 0$ and $\sqrt{d} = \det_{k,l}(\omega_k^{(l)}) > 0$. Now fix a prime number p and an embedding of the algebraic closure of $\mathscr{K}$ into $\mathbf{C}_p$, the completion of an algebraic closure of the p-adic field $\mathbf{Q}_p$. The p-adic regulator of $\mathscr{K}$ is defined by $R_p(\mathscr{K}) = \det_{i,j}(\log_p(\varepsilon_i^{(j)}))$. Finally let $h_{\mathscr{K}}$ be the class number of $\mathscr{K}$ and $L_p(s, \chi)$ be the p-adic L-function for the primitive Dirichlet character χ.

Theorem 2.1 (p-adic class number formula). *For the values of $R_p(\mathscr{K})$, h_k, and $L_p(1, \chi)$ defined above, we have*

$$2^{n-1} h_{\mathscr{K}} \left(\frac{R_p(\mathscr{K})}{\sqrt{d}} \right) = \prod_{\chi \neq 1} (1 - \chi(p)p^{-1})^{-1} L_p(1, \chi).$$

Corollary. *Assume $(p, 2d) = 1$, and $p^{n-1}/R_p(\mathscr{K})$ is integral. Then the following congruence holds between integral elements of $\mathbf{C}_p$:*

$$h_{\mathscr{K}} \equiv \frac{p^{n-1}\sqrt{d}}{R_p(\mathscr{K})2^{n-1}} \prod_{\chi \neq 1} \chi(p) \left[\prod_{\chi \neq 1} \left(\frac{1}{d} \sum_{\substack{1 \leq a \leq p\,d \\ (a,p)=1}} \chi(a) \left(\frac{1 - a^{p-1}}{p} \right) \right) \right] \pmod{p}.$$

Proof. From the theorem, we immediately have

$$h_{\mathscr{K}} \equiv \frac{p^{n-1}\sqrt{d}}{R(\mathscr{K})2^{n-1}} \prod_{\chi \neq 1} \left(\frac{L_p(1, \chi)}{p - \chi(p)} \right).$$

For each $\chi \neq 1$, the term $L_p(1, \chi)/(p - \chi(p))$ is p-integral when $(p, 2d) = 1$, as seen in [9, p. 60]. More specifically, this reference shows that this term is congruent to

$$-\chi(p)L_p(1, \chi) \equiv -\chi(p)L_p(0, \chi) \equiv -\chi(p) \left(\frac{-1}{d} \sum_{\substack{1 \leq a \leq p\,d \\ (a,p)=1}} \chi(a) \left(\frac{\log_p(a)}{p} \right) \right)$$

$$\equiv \chi(p) \left(\frac{1}{d} \sum_{\substack{1 \leq a \leq p\,d \\ (a,p)=1}} \chi(a) \left(\frac{1 - a^{p-1}}{p} \right) \right) \pmod{p}.$$

The last congruence follows from

$$\log_p(a) = \frac{\log_p(a^{p-1})}{p-1} \equiv -\log_p(a^{p-1}) \equiv 1 - a^{p-1} \pmod{p^2},$$

since $a^{p-1} \equiv 1 \mod p$. Multiplying over all $\chi \neq 1$ yields the result. □

Now let $\mathscr{K}$ be a real quadratic field with discriminant $d > 0$, and assume from now on that p is an odd prime not dividing d. There is only one character

$\chi \neq 1$, given by $\chi(a) = (d/a)$. Let the fundamental unit be $\varepsilon_1 = (t + u\sqrt{d})/2$, chosen so that $\varepsilon_1 > 1$ in order to have $R(\mathscr{K}) = \log(\varepsilon_1) > 0$. The norm of the fundamental unit is $N(\varepsilon_1) = \pm 1$.

The following result is well known.

Lemma 2.1. *Write* $\varepsilon_1^{p-\chi(p)} = (T + U\sqrt{d})/2$ *with* T, $U \in \mathbf{Z}$. *Then* $U \equiv 0$ *and* $T \equiv 2N(\varepsilon_1)^{(\chi(p)-1)/2} \pmod{p}$.

Proof. First,

$$\varepsilon_1^p = \left(\frac{t + u\sqrt{d}}{2}\right)^p \equiv (t + \chi(p)u\sqrt{d})/2 \mod p.$$

When $\chi(p) = -1$, we have

$$\frac{(T + U\sqrt{d})}{2} \equiv \frac{(t - u\sqrt{d})}{2}\frac{(t + u\sqrt{d})}{2} \pmod{p},$$

and this leads to the congruences $T \equiv 2N(\varepsilon_1)$, $U \equiv 0 \pmod{p}$.

When $\chi(p) = 1$, the result follows from $\varepsilon_1^p \equiv \varepsilon_1 \pmod{p}$. □

Now we compute $R_p(\mathscr{K}) = \log_p(\varepsilon_1^{p-\chi(p)})/(p - \chi(p))$. The lemma allows us to use the power series for $\log_p$ in the computation:

$$\log_p\left(\frac{T}{2} + \frac{U\sqrt{d}}{2}\right) = \log_p(1 + UT^{-1}\sqrt{d}) + \log_p\left(\frac{T}{2}\right)$$
$$\equiv UT^{-1}\sqrt{d} + N(\varepsilon_1)^{(\chi(p)-1)/2}\frac{T}{2} - 1 \pmod{p^2}.$$

Hence $R_p(\mathscr{K})/p$ is integral and congruent to

$$-\chi(p)\left(\frac{U}{p}T^{-1}\sqrt{d} + \frac{N(\varepsilon_1)^{(\chi(p)-1)/2}T/2 - 1}{p}\right) \pmod{p}.$$

Multiplying the congruence in the corollary by this factor and a factor of $2\chi(p)\sqrt{d}$, we have

$$-2h_{\mathscr{K}}\left(\frac{U}{p}t^{-1}d + \left(\frac{N(\varepsilon_1)^{(\chi(p)-1)/2}T/2 - 1}{p}\right)\sqrt{d}\right)$$
$$\equiv \sum_{\substack{1 \le a \le p\,d \\ (a,p)=1}} \left(\frac{1 - a^{p-1}}{p}\right) \pmod{p},$$

when $R_p(\mathscr{K})/p$ is a p-unit. When $R_p(\mathscr{K})/p$ is not a p-unit, the congruence also holds, but is not very useful. Notice that this is now a congruence in $\mathscr{K}$,

so we may conclude that

$$-2h_{\mathfrak{K}}\left(\frac{U}{p}T^{-1}d\right) \equiv \sum_{\substack{1\le a\le p\,d\\(a\,,\,p)=1}} \chi(a)\left(\frac{1-a^{p-1}}{p}\right) \pmod p$$

(and $(N(\varepsilon_1)^{(\chi(p)-1)/2}T/2-1)/p \equiv 0$).

Next we will simplify the sum on the right-hand side of the congruence.

Let $L(a) = ((1-a^{p-1})/p)$ when $(a\,,\,p) = 1$, and $L(a) = 0$ otherwise. Clearly

$$L(a) \equiv L(a') \pmod p \quad \text{for } a \equiv a' \pmod{p^2}.$$

Since $L(a) \equiv (\log_p(a)/p) \pmod p$, we also have $L(ac) \equiv L(a) + L(c) \pmod p$, for $(ac\,,\,p) = 1$. Putting $\mathscr{L}(b) = \sum_{k=0}^{p-1} L(b + k\,d)$ yields

$$-2h_{\mathfrak{K}}\left(\frac{U}{p}T^{-1}d\right) \equiv \sum_{b=1}^{d} \chi(b)\mathscr{L}(b) \pmod p. \tag{2.1}$$

Lemma 2.2. *If* $b \equiv b' \mod p$, *then* $\mathscr{L}(b) \equiv \mathscr{L}(b') \pmod p$.

Proof. First,

$$\mathscr{L}(b) \equiv L\left(\prod_{\substack{0\le k\le p-1\\(p\,,\,b+k\,d)=1}} (b + k\,d)\right) \pmod p,$$

by the logarithmic property of L. Now

$$\prod_{\substack{0\le k\le p-1\\(p\,,\,b+k\,d)=1}} (X + (b + k\,d)) \equiv X^{p-1} + \prod_{\substack{0\le k\le p-1\\(p\,,\,b+k\,d)=1}} (b + k\,d) \pmod{pX}$$

in $\mathbf{Z}[X]$. If $b' = b + rp$, set $X = rp$ to obtain

$$\prod_{\substack{0\le k\le p-1\\(p\,,\,b+k\,d)=1}} (b' + k\,d) \equiv \prod_{\substack{0\le k\le p-1\\(p\,,\,b+k\,d)=1}} (b + k\,d) \pmod{p^2}.$$

The result follows. □

Corollary. *Let* $c = c(b)$ *be the smallest positive integer such that* $b \equiv cd \pmod p$. *Then*

$$\mathscr{L}(b) \equiv L\left(\prod_{\substack{0\le k\le p-1\\(p\,,\,c+k)=1}} (c + k)\right) - L(d) \pmod p.$$

Proof. We have

$$\mathscr{L}(b) \equiv \mathscr{L}(cd) = \sum_{\substack{0\le k\le p-1\\(p,\,cd+kd)=1}} L(cd+kd) \equiv \sum_{\substack{0\le k\le p-1\\(p,\,c+k)=1}} (L(c+k)+L(d))$$

$$\equiv L\left(\prod_{\substack{0\le k\le p-1\\(p,\,c+k)=1}} (c+k)\right) - L(d) \pmod p. \quad \square$$

The next result is easily derived from the work of Glaisher [2]; however, since there is a simple, self-contained proof, we present it here.

Proposition 2.1. *If p is a prime, then*

$$\prod_{\substack{0\le k\le p-1\\(p,\,c+k)=1}} (c+k) \equiv (p-1)!\left(1+p\sum_{1\le j\le c-1}\frac{1}{j}\right) \pmod{p^2}.$$

Proof. We have

$$\prod_{\substack{0\le k\le p-1\\(p,\,c+k)=1}} (c+k) = \frac{(p-1)!}{(c-1)!}(p+1)(p+2)\cdots(p+c-1)$$

$$\equiv \frac{(p-1)!}{(c-1)!}\left((c-1)!+p\sum_{1\le j\le c-1}\frac{(c-1)!}{j}\right)$$

$$= (p-1)!\left(1+p\sum_{1\le j\le c-1}\frac{1}{j}\right) \pmod{p^2}. \quad \square$$

Define

$$S = -\sum_{1\le b\le d-1}\chi(b)\left(\sum_{1\le j\le c(b)-1}\frac{1}{j}\right) \pmod p.$$

Then we may summarize our results as follows.

Theorem 2.2. *For the values of U, T, and S defined above we have*

$$2h_{\mathscr{K}}\left(\frac{U}{p}T^{-1}d\right) \equiv S \pmod p.$$

Proof. By (2.1),

$$-2h_{\mathcal{K}}\left(\frac{U}{p}T^{-1}d\right) \equiv \sum_{b=1}^{d}\chi(b)\mathcal{L}(b)$$

$$\equiv \sum_{b=1}^{d}\chi(b)\left[L\left(\prod_{\substack{0\le k\le p-1\\(p,c+k)=1}}(c+k)\right)-L(d)\right]$$

(by the Corollary to Lemma 2.2)

$$= \sum_{b=1}^{d}\chi(b)\left[L\left(\prod_{\substack{0\le k\le p-1\\(p,c+k)=1}}(c+k)\right)\right]$$

$$\equiv \sum_{b=1}^{d}\chi(b)L\left((p-1)!\left(1+p\sum_{1\le j\le c-1}\frac{1}{j}\right)\right)$$

(by Proposition 2.1)

$$\equiv \sum_{b=1}^{d}\chi(b)\left[L((p-1)!)+L\left(1+p\sum_{1\le j\le c-1}\frac{1}{j}\right)\right]$$

$$\equiv \sum_{b=1}^{d}\chi(b)\left[L\left(1+p\sum_{1\le j\le c-1}\frac{1}{j}\right)\right] \pmod{p}.$$

Since $L(1+pk)\equiv k \pmod{p}$, we have our result. □

3. Algorithm to determine $T^{-1}U/p \pmod{p}$

Define $t_k, u_k \in \mathbf{Z}$ by $t_1 = t$, $u_1 = u$, and

$$\frac{t_k + u_k\sqrt{d}}{2} = \varepsilon_1^k = \left(\frac{t+u\sqrt{d}}{2}\right)^k. \tag{3.1}$$

If we put $m = p - \chi(p)$, then $T = t_m$, $U = u_m$. In this section we will show how the value of $T^{-1}U/p \pmod{p}$ can be efficiently computed.

We first consider the continued fraction expansion of $(d+\sqrt{d})/2$. We put[1] $P_0 = d$, $Q_0 = 2$, $q_0 = [(d+\sqrt{d})/2]$, and define

$$P_{i+1} = q_iQ_i - P_i,$$
$$Q_{i+1} = (d - P_{i+1}^2)/Q_i,$$
$$q_{i+1} = [(P_{i+1}+\sqrt{d})/Q_{i+1}] \qquad (i = 0, 1, 2, \ldots).$$

[1] We use $[\alpha]$ to denote the integer part of α.

There must be some minimal $j \geq 1$ such that either $P_{j+1} = P_j$ or $Q_{j+1} = Q_j$ (see, for example, Theorem 2.2 of Stephens and Williams [7]). In the first case we have $N(\varepsilon_1) = 1$, and in the second $N(\varepsilon_1) = -1$. Also, if $B_{-2} = 1$, $B_{-1} = 0$, and we define $B_{i+1} \equiv q_{i+1}B_i + B_{i-1} \pmod{p^2}$, we get

$$u \equiv B_{j-1}(B_{j-2} + B_j) \pmod{p^2},$$
$$t \equiv Q_{j-1}B_{j-1}^2 + Q_jB_{j-2}^2 + uP_j \pmod{p^2}$$

when $N(\varepsilon_1) = 1$, or

$$u \equiv B_j^2 + B_{j-1}^2 \pmod{p^2},$$
$$t \equiv P_ju + Q_{j-1}B_{j-1}B_j + Q_jB_{j-1}B_{j-2} + (P_{j+1} - P_j)B_jB_{j-2} \pmod{p^2}$$

when $N(\varepsilon_1) = -1$.

Thus, by making use of the continued fraction algorithm, we can easily compute $N(\varepsilon_1)$ and the values of t and $u \pmod{p^2}$. This process is of complexity $O(d^{1/2+\varepsilon})$.

We next show how to compute $U_m/p \pmod{p}$. By using (3.1) we can easily prove the following simple identities:

$$t_{2n} = t_n^2 - 2N(\varepsilon_1)^n, \tag{3.2}$$

$$t_{2(n+s)} = t_{2n}t_{2s} - t_{2(n-s)}, \tag{3.3}$$

$$dutu_n = 2t_{n+2} - t_2t_n. \tag{3.4}$$

Let $(b_0b_1b_2\cdots b_k)_2$ be the binary representation of $m/2$ $(b_j = 0, 1; j = 0, 1, 2, \ldots, k)$. Put[2] $\mathscr{P}_0 = \{t_2, t_4\} \pmod{p^2}$ and deduce $\mathscr{P}_{i+1}$ from $\mathscr{P}_i = \{A, B\} \pmod{p^2}$ by

$$\mathscr{P}_{i+1} = \begin{cases} \{A^2 - 2, AB - t_2\} \pmod{p^2} & \text{when } b_{i+1} = 0, \\ \{AB - t_2, B^2 - 2\} \pmod{p^2} & \text{when } b_{i+1} = 1. \end{cases}$$

Note that if we put $s_0 = b_0 = 1$ and $s_{i+1} = 2s_i + b_{i+1}$, we can easily prove from (3.2) and (3.3) that

$$\mathscr{P}_i = \{t_j, t_{j+2}\} \pmod{p^2},$$

where $j = 2s_i$. Thus, $\mathscr{P}_k = \{t_m, t_{m+2}\} \pmod{p^2}$. From (3.4) we find that

$$dutu_m/p \equiv (2t_{m+2} - t_2t_m)/p \pmod{p}. \tag{3.5}$$

[2]We use $\{a, b\} \pmod{m}$ to denote the set of residues $\{a \pmod{m}, b \pmod{m}\}$.

By (3.5), Theorem 2.2, and Lemma 2.1 we have

$$h_{\mathscr{R}} \equiv v_m^{-1} utN(\varepsilon)^{(\chi-1)/2} S \pmod{p}, \tag{3.6}$$

where $\chi = \chi(p)$,

$$v_m \equiv (2t_{m+2} - t_2 t_m)/p \pmod{p},$$

and $p \nmid v_m$. If $p | v_m$, then another value of p must be selected and the algorithm executed again. Notice that the complexity of evaluating $v_m \pmod{p}$ (given u and t) is $O(m^{\varepsilon})$.

4. Some results concerning $S \pmod{p}$

Define

$$W(b) \equiv \sum_{j=1}^{c(b)-1} \frac{1}{j} \pmod{p}. \tag{4.1}$$

By Theorem 2.2 we know that

$$S = -\sum_{b=1}^{d-1} \chi(b) W(b) \pmod{p}. \tag{4.2}$$

Unfortunately, the evaluation of S is a process of complexity $O(d^{1+\varepsilon})$. While we are unable to improve upon this complexity measure for evaluating $S \pmod{p}$, we can improve somewhat the process implied by using (4.2) to compute $S \pmod{p}$. We will do this by evaluating the sum in a different order than that indicated by (4.2)—an order that will eliminate the need for determining $\chi(b)$ by using the expensive (in our application) quadratic reciprocity technique. In order to do this, we need to prove some simple results concerning S.

Let $d = 2^{\lambda} Q$, where $2 \nmid Q$. We can only have $\lambda = 0, 2, 3$ and $Q = \prod_{i=1}^{k} q_i$, where the q_i $(i = 1, 2, \ldots, k)$ are distinct odd primes. Let g_i be a primitive root of q_i and define h_i by

$$h_i \equiv g_i \pmod{q_i}, \qquad h_i \equiv 1 \pmod{d/q_i}.$$

If $\lambda \neq 3$, put $h_0 = 1$; otherwise set

$$h_0 \equiv 5 \pmod{8}, \qquad h_0 \equiv 1 \pmod{Q}.$$

We are now able to prove

Lemma 4.1. *Let $\mathscr{R}$ be a reduced system of residues modulo d. If $r \in \mathscr{R}$, then*

$$r \equiv (-1)^{\alpha} \prod_{i=0}^{k} h_i^{\beta_i} \pmod{d}, \tag{4.3}$$

where $0 \le \beta_0 \le 1$, $0 \le \beta_i \le q_i - 2$ $(i = 1, 2, 3, \ldots, k)$, $\alpha = 0$ *when* $\lambda = 0$, *and* $0 \le \alpha \le 1$ *when* $\lambda > 0$.

Proof. Let $\mathscr{S}$ be the set of all distinct values of r modulo d which satisfy (4.3). If r, r' are given by (4.3), then

$$r \equiv (-1)^{\alpha} \prod_{i=0}^{k} h_i^{\beta_i} \pmod{d}, \qquad r' \equiv (-1)^{\alpha'} \prod_{i=0}^{k} h_i^{\beta_i'} \pmod{d}.$$

If $r \equiv r' \pmod{d}$, it is easy to verify from the construction of the h_i values that $\alpha \equiv \alpha' \pmod 2$, $\beta_0 \equiv \beta_0' \pmod 2$, and $\beta_i \equiv \beta_i' \pmod{q_i - 1}$ $(i = 1, 2, \ldots, k)$. Thus, $|\mathscr{S}| = \phi(d)$. Since all the elements of $\mathscr{S}$ are relatively prime to d, we must have $\mathscr{R} = \mathscr{S}$. □

Now $\chi(r) = \chi(d - r)$; hence, if r is given by (4.3), we get

$$\chi(r) = \prod_{i=0}^{k} \chi(h_i)^{\beta_i}.$$

Also,

$$\chi(h_i) = \left(\frac{2}{h_i}\right)^{\lambda} (-1)^{((Q-1)/2)((h_i-1)/2)} \left(\frac{h_i}{Q}\right)$$

and

$$\left(\frac{h_i}{Q}\right) = \left(\frac{h_i}{q_i}\right)\left(\frac{h_i}{Q/q_i}\right) = \left(\frac{h_i}{q_i}\right) = -1 \qquad (1 \le i \le k).$$

If $\lambda \ne 0$, then $h_i \equiv 1 \pmod 4$ and

$$\chi(h_i) = \left(\frac{2}{h_i}\right)^{\lambda} \left(\frac{h_i}{Q}\right).$$

If $\lambda = 3$, then

$$\chi(h_i) = \left(\frac{2}{h_i}\right)\left(\frac{h_i}{Q}\right) = \begin{cases} (h_i/Q), & 1 \le i \le k, \\ (2/h_0), & i = 0. \end{cases}$$

Thus, no matter what the value of λ, we get $\chi(h_i) = -1$ $(i = 0, 1, 2, \ldots, k)$. It follows that if r is given by (4.3), then

$$\chi(r) = (-1)^t, \tag{4.4}$$

where $t = \sum_{i=0}^{k} \beta_i$.

We now need

Lemma 4.2. *For* $W(b)$ *defined by* (4.1) *we have* $W(d - b) \equiv W(b) \pmod p$.

Proof. Put $c' = c(d-b)$, $c = c(b)$; then $c' = c(d-b) \equiv d^{-1}(d-b) \equiv 1 - d^{-1}b \equiv 1 - c \pmod{p}$; hence, $c' = p + 1 - c$, and we have

$$W(d-b) \equiv \sum_{j=1}^{p-c} \frac{1}{j} \pmod{p}.$$

Now

$$\sum_{j=1}^{p-c} \frac{1}{j} + \sum_{j=p-c+1}^{p-1} \frac{1}{j} \equiv 0 \pmod{p}.$$

It follows, then, that

$$W(d-b) \equiv -\sum_{j=p-c+1}^{p-1} \frac{1}{j} \equiv -\sum_{i=1}^{c-1} \frac{1}{p-i} \equiv W(b) \pmod{p}. \quad \square$$

Define $\mathscr{R}_1$ to be the set of all the distinct values of $r \pmod{d}$ given by (4.3) when we fix $\alpha = 0$, and let $\mathscr{R}_2$ be the set of all the distinct values of $r \pmod{d}$ given by (4.3) when we fix $\alpha = 1$. From Lemma 4.2 we can deduce

Theorem 4.1. *If* $\lambda > 0$, *then* $S \equiv -2\sum_{r\in\mathscr{R}_1} W(r)\chi(r) \pmod{p}$.

Proof. By Lemma 4.1 we know that

$$S \equiv -\sum_{r\in\mathscr{R}_1} W(r)\chi(r) - \sum_{r\in\mathscr{R}_2} W(r)\chi(r) \pmod{p}.$$

Now $r \in \mathscr{R}_1$ if and only if $d - r \in \mathscr{R}_2$; hence,

$$\begin{aligned} S &\equiv -\sum_{r\in\mathscr{R}_1} W(r)\chi(r) - \sum_{r\in\mathscr{R}_1} W(d-r)\chi(d-r) \pmod{p} \\ &\equiv -2\sum_{r\in\mathscr{R}_1} W(r)\chi(r) \pmod{p} \end{aligned}$$

by Lemma 4.2. $\square$

Put $T \equiv \sum_{r\in\mathscr{R}_1} W(r)\chi(r) \pmod{p}$. Since $S \equiv -T \pmod{p}$ when $\lambda = 0$, and $S \equiv -2T \pmod{p}$ when $\lambda > 0$, we see that we can easily evaluate $S \pmod{p}$ once we know $T \pmod{p}$.

In the next section we provide a simple computer algorithm for calculating $T \pmod{p}$.

5. Computer implementation and results

By using some of the results derived in §4, we can now present our algorithm for determining $T \pmod{p}$.

Algorithm 5.1.

(1) Input d, p, k, h_0, h_i, q_i $(i = 1, 2, 3, \ldots, k)$.

(2) Put $i \leftarrow k$, $r \leftarrow 1$, $T \leftarrow W(1) \pmod p$, $\chi \leftarrow 1$,
$b_0 \leftarrow 1$, $j_0 \leftarrow 0$, $b_i \leftarrow q_i - 2$, $j_m \leftarrow 0$ $(m = 1, 2, 3, \ldots, k)$.

(3) Perform the following steps as long as $i \geq 0$:

(a) If $j_i < b_i$, then put
$j_i \leftarrow j_i + 1$
$r \leftarrow rh_i \pmod d$
(*) $T \leftarrow T + \chi W(r) \pmod p$
$\chi \leftarrow -\chi$
$i \leftarrow k$
go to (a)

(b) If $j_i = b_i$, put
$j_i \leftarrow 0$
$\chi \leftarrow -\chi$
$r \leftarrow rh_i \pmod d$
$i \leftarrow i - 1$.

(4) Output T.

This algorithm is correct because b_i is odd $(i = 0, 1, 2, \ldots, k)$ and $h_i^{-b_i} \equiv h_i \pmod d$; thus, we see that each value of r and χ used in (*) is such that $r \equiv \prod_{i=0}^{k} h_i^{j_i} \pmod d$ and $\chi = (-1)^t$, where $t = \sum_{i=0}^{k} j_i$, for the $(k+1)$-tuple $(j_0, j_1, \ldots, j_k)$ in memory at the time (*) is evaluated. Furthermore, at (*) the $(k+1)$-tuple $(j_0, j_1, \ldots, j_k)$ $(0 \leq j_i \leq b_i;\ i = 0, 1, 2, \ldots, k)$ will take on each possible set of values exactly once.

We have not yet considered the problem of evaluating $W(r)$ in Algorithm 5.1. We define $Y(j) \equiv \sum_{i=1}^{j-1} 1/i \pmod p$. It is clear that

$$W(n) \equiv Y(j) \pmod p \tag{5.1}$$

when $j \equiv n d^{-1} \pmod p$. Thus, in order to evaluate $W(r)$, we first tabulate and store in memory (in a precomputation process antecedent to Algorithm 5.1) all the values of $Y(j) \pmod p$ $(j = 0, 1, 2, \ldots, p-1)$. Notice that the values of $Y(j)$ are independent of the value of d_i, hence, if a large table of class numbers is to be produced, we need only compute the table of $Y(j)$'s once for each p. For a given d value we use this table and (5.1) to produce a table of values for $W(n)$ $(n = 0, 1, \ldots, p-1)$. To evaluate $W(r)$, then, simply involves a table look-up once r has been reduced modulo p.

All of the algorithms discussed here were implemented in Assembly Language on a Micro VAX II Computer in the Department of Computer Science at the

University of Manitoba. During the course of running this program, it was discovered that the part of the loop which was used to compute T (the computation of T was by far the most expensive of the routines) which was most time consuming was the evaluation of $r \pmod{d}$. We improved this process by using the following technique.

We select a value of $w \in \mathbf{Z}$ such that $2^w < \sqrt{d} < 2^{w+1}$. For each h_i we tabulate

$$\mathscr{X}_i = \{X_i(j) | j = 0, 1, 2, \ldots, [d/2^w]\}$$

and

$$\mathscr{Z}_i = \{Z_i(j) | j = 0, 1, 2, \ldots, 2^w - 1\}.$$

Here, $X_i(j)$ is the least positive residue of $2^w h_i j \pmod{d}$, and $Z_i(j)$ is the least positive residue of $h_i j \pmod{d}$. This tabulation is also done in the precomputation phase described earlier. Each of these tables has about $\sqrt{d}$ entries, and there are $2k = O(\log d)$ of them. For each r it is easy to find q, s such that $r = 2^w q + s$ $(0 \le s < 2^w)$. Since $r < d$, we have $q < [d/2^w]$; also, $h_i r = 2^w q h_i + s h i$; hence, the value of $h_i r$ reduced modulo d is either

$$n = X_i(q) + Z_i(s) \quad (n < d) \qquad \text{or} \qquad n - d \quad (n > d).$$

This table look-up process eliminates the need to do the expensive multiplication (by h_i) and division (by d) to obtain the new r value.

Our program was run for all square-free values of $D < 10^6$; a table, similar in format to that of Wada [8], has been deposited in the UMT File. This table gives the values of $h_{\mathscr{K}}$ and $N(\varepsilon_1)$ for each square-free $D < 10^6$. For values of p we used 331, 337, 347, 349, and 353. Extra values of p were needed in those cases in which we found that the first prime (or primes) actually divided v_m in (3.6), making the determination of v_m^{-1} impossible. For example, the first prime used (331) divided v_m about once in every 100 values of D considered. Also, since the evaluation of v_m is cheap compared to that of T, we always computed v_m first in order to find a value of p such that $p \nmid v_m$. For small values of D the method executed quite rapidly; for example, only 18.5 minutes of CPU time were needed to find all the class numbers for all the D values such that $D < 10^4$. However, to find the class numbers for all the values of D between 998001 and 1000000 required about 10.5 hours of CPU time. In fact, near the upper end of our range it could take as long as 35–40 seconds to find $h_{\mathscr{K}}$ for certain values of D. Of this time only about a fraction of a second was spent in finding $v_m \pmod{p}$.

For the purpose of comparison we mention that the algorithm of Lerch, as implemented by Williams and Broere (see [4] for references), computed the values for $h_{\mathscr{K}}$ for all values of $D < 150,000$ in about 7 hours of CPU time on an IBM 370-158 computer. (This is the largest table in existence for which there is some published account; A. O. L. Atkin (see [4]) is said to have produced a table up to 4×10^6.) To accomplish this same task, our algorithm requires 62 hours on the Micro VAX II. Although it is difficult to compare different

machines, it is fair to say that the speed of the arithmetic of the IBM 370-158 is less than 9 times the speed of the Micro VAX II; thus, we see that even up to 150,000 the method of Lerch is more efficient than our method. Also, since the Lerch technique is of complexity $O(D^{1/2+\varepsilon})$, it would be even better for larger values of D. However, we should mention here that our algorithm, unlike the Lerch algorithm, is quite simple to program and does not require any approximations to transcendental functions or floating-point arithmetic. In conclusion, we see that this technique works reasonably well for small values of D, or even isolated larger values of D; however, for computing large tables it is much slower than the methods mentioned in [4].

Bibliography

1. Y. Amice and J. Fresnel, *Fonctions zêta p-adiques des corps de nombres abeliens réels*, Acta Arith. **20** (1972), 353–384.
2. J. W. L. Glaisher, *Residue of the product of p numbers in arithmetical progression mod p^2 and p^3*, Messenger Math. **30** (1900–01), 71–92.
3. H. W. Leopoldt, *Eine p-adische Theorie der Zetawerte.* II. *Die p-adische T-Transformation*, J. Reine Angew. Math. **274/275** (1975), 224–239.
4. R. A. Mollin and H. C. Williams, *Computation of the class number of a real quadratic field*, Advances in the Theory of Computation and Computational Mathematics (to appear).
5. I. S. Slavutskii, *Upper bounds and numerical calculation of the number of ideal classes of real quadratic fields*, Amer. Math. Soc. Transl. (2) **82** (1969), 67–72.
6. R. G. Stanton, C. Sudler, Jr., and H. C. Williams, *An upper bound for the period of the simple continued fraction for $\sqrt{d}$*, Pacific J. Math. **67** (1976), 525–536.
7. A. J. Stephens and H. C. Williams, *Some computational results on a problem concerning powerful numbers*, Math. Comp. **50** (1988), 619–632.
8. H. Wada, *A table of ideal class numbers of real quadratic fields*, Kôkyûroku in Math., no. 10, Sophia University, Tokyo, 1981.
9. L. C. Washington, *Introduction to cyclotomic fields*, Springer-Verlag, New York, 1982.

Fachbereich 10, Informatik, Universität des Saarlandes, D-6600 Saarbrücken, Federal Republic of Germany

Department of Mathematics and Statistics, University of Vermont, Burlington, Vermont 05405

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